Personal Computer für die Praxis

computer persönlich

Rund um TI 99/4: mehr als nur ein Heimcomputer

Was taugt der lang erwartete TI-Writer?
Mit diesem Computer macht das Lernen Spaß

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

Halbvollständige Sammlung von Auflistungen und Neuigkeiten von 1982-1984 für den ti99/4a

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

TI99/4A Collection

Hierzu sind folgende Schritte notwendig; nachdem man Track 17/Sektor Ø eingelesen hat:
a) Positionieren des Cursors auf das erste Zeichen der zweiten Zeile.
b) Von hier aus acht Bytes zurückgehen.
c) B)yte Modus.
d) Folgende zwölf Bytes eingeben:
FF ØØ FF ØØ FF ØØ
FF ØØ FF ØØ FF ØØ
e) Q)uit
f) W)rite

Damit stehen auch die vom DOS belegten Sektoren zur Verfügung.

Eine solchermaßen geänderte Diskette läßt sich selbstverständlich nicht mehr booten, sondern kann nur verwendet werden, nachdem das System mit einer anderen Diskette hochgefahren worden ist. Dafür bietet sie eben mehr Speicherplatz.

Abschließend noch ein paar Worte zu den beiden Maschinensprache-routinen. DISP stellt den Inhalt eines eingelesenen Sektors auf dem Bildschirm dar und RWTS versorgt die DOS-RWTS-Routinen mit den nötigen Parametern.

(Dipl.-Phys. Peter-M. Oden)

Disketten-Inhaltsverzeichnis

Besitzt man einen Drucker, dann ist es sinnvoll, Disketten-Inhaltsverzeichnisse auszudrucken und der Diskette beizulegen.

```
100 CALL CLEAR
110 DIM TYPE$(5)
120 TYPE$(1)="DIS/FIX"
130 TYPE$(2)="DIS/VAR"
140 TYPE$(3)="INT/FIX"
150 TYPE$(4)="INT/VAR"
160 TYPE$(5)="PROGRAM"
170 PRINT "MASTER DISK (1-3)":
180 CALL KEY(0,I,_)
190 IF (I<49)+(I>51)THEN 180
200 I=I-48
210 CALL CLEAR
220 PRINT "1 DRUCKER":"2 KEIN DRUCKER"
230 CALL KEY(0,KEY,STA)
240 IF (KEY<49)+(KEY>50)THEN 230
250 CALL CLEAR
260 IF KEY=50 THEN 280
270 OPEN #2:"RS232",DISPLAY
280 OPEN #1:"DSK"&STR$(I)&".",INPUT ,RELATIVE,INTERNAL
290 INPUT #1:A$,J,J,K
300 DISPLAY ::"DSK"&STR$(I),A$:"FREI=";K;"BELEGT=";J-K
310 IF KEY=50 THEN 330
320 PRINT #2:"DSK"&STR$(I),"DISKNAME";A$:"FREI=";K,"BELEGT=";J-K
330 DISPLAY :" FILENAME  SIZE    TYPE    P":"---------- ---- ---------- -";
340 IF KEY=50 THEN 360
350 PRINT #2:" FILENAME  SIZE    TYPE    P":"---------- ---- ---------- -";
360 INPUT #1:A$,I,J,K
370 IF LEN(A$)=0 THEN 510
380 DISPLAY :A$;TAB(12);J;TAB(17);TYPE$(ABS(I));
390 IF KEY=50 THEN 410
400 PRINT #2::A$;TAB(12);J;TAB(17);TYPE$(ABS(I));
410 IF ABS(I)=5 THEN 460
420 B$="  "&STR$(K)
430 DISPLAY SEG$(B$,LEN(B$)-2,3);
440 IF KEY=50 THEN 460
450 PRINT #2:SEG$(B$,LEN(B$)-2,3);
460 IF I>0 THEN 360
470 DISPLAY TAB(28);"Y";
480 IF KEY=50 THEN 500
490 PRINT #2:TAB(28);"Y";
500 GOTO 360
510 CLOSE #1
520 IF KEY=50 THEN 540
530 CLOSE #2
540 PRINT ::"WEITER (J/N) ?"
550 CALL KEY(0,KEY,STA)
560 IF KEY=74 THEN 100
570 IF KEY<>78 THEN 550
580 END
```

Bild 1. Listing Disketten-Inhaltsverzeichnis

100	Löscht den Bildschirm
110	Hält Speicherplatz frei
120 - 160	Dieses sind die Arten der Files
180	Eingabe einer Zahl
190	Bei falscher Eingabe nach 180
200	I wird eine Zahl von 1-3
210	Bildschirm löschen
230	Eingabe einer Zahl
240	Sicherung
250	Bildschirm löschen
260	Bei nicht vorhandenem Drucker nach 2.o
270	Ausgabeeröffnung
280	Eingabeeröffnung für Diskette
290	Zeichen werden von der Diskette geholt
300	Ausgabe der freien bzw. belegten Sektoren auf dem Bildschirm
320	Dasselbe wie bei 300, nur auf dem Drucker
360	Zeichen von Diskette holen
370	Wenn kein File mehr vorhanden, dann nach 510
410	Bei Type Programm nach 460
420 - 450	Hier entsteht die fixierte bzw. die variable Länge der Datensätze
510 - 530	Die Ausgabekanäle werden geschlossen

Erläuterungen zum Programm (Bild 3)

```
DSK1          DISKNAMEMAGRODISK1
FREI= 0       BELEGT= 358
 FILENAME  SIZE    TYPE      P
---------- ---- ---------- -
ALIEN       23   PROGRAM
BALL        4    PROGRAM
BANDIT      13   PROGRAM
BASIC       5    PROGRAM
BIO         42   PROGRAM
```

Bild 2. So sieht's aus!

TI 99/4A

So kann man mit einem Blick überschauen, welche Dateien auf einer Diskette vorhanden sind und braucht sie nicht erst ins Laufwerk einzulegen.

Dazu dient das kleine Hilfsprogramm. Es druckt nicht nur die Dateinamen, sondern auch die Dateigröße und Dateiart, sowie die Zahl der freien und belegten Blöcke auf der Disk.

(Martin Grotegut)

Einfache Kartei

Auch mit einem Kassettenrecorder als Speichermedium ist es möglich, Karteien zu verwalten — vorausgesetzt sie sind nicht zu umfangreich. Hier ein kleines Karteiprogramm, das sich gut für Adressen, Telefonnummern oder als elektronischer Terminkalender eignet.

TI 99/4(A)

```
10 DIM D$(5,200)
20 CALL CLEAR
30 PRINT "PERSONEN-DATEI":"-----
---------":::"VON":" TORSTEN NIE
MIETZ":" MARBACHER WEG 39":" D-2
800  BREMEN 1"
40 PRINT :::"WAHLLISTE:":"1- DAT
EN LADEN":"2- DATEN SUCHEN":"3-
DATEN SPEICHERN":"4- NEUE DATEN
EINGEBEN"
50 PRINT "5- KATALOG":"6- ENDE":
::
60 CALL SOUND(150,1400,5)
70 CALL KEY(0,K,S)
80 IF S=0 THEN 70
90 IF K<49 THEN 70
100 IF K>54 THEN 70
110 CALL CLEAR
120 ON K-48 GOTO 130,220,720,810
,670,900
130 OPEN #1:"CS1",INTERNAL,INPUT
 ,FIXED
140 INPUT #1:A
150 FOR D=1 TO A
160 FOR E=1 TO 5
170 INPUT #1:D$(E,D)
180 NEXT E
190 NEXT D
200 CLOSE #1
210 GOTO 20
220 INPUT "BITTE EINE ZEILE EING
EBEN.  ":Z$
230 FOR I=1 TO A
240 FOR J=1 TO 5
250 IF Z$=D$(J,I)THEN 320
260 NEXT J
270 NEXT I
280 PRINT ::Z$:"IST NICHT ZU FIN
DEN!"
290 CALL SOUND(200,219,0)
300 INPUT "BITTE -ENTER- DRUECKE
N.  ":Z$
310 GOTO 20
320 PRINT "DATEINUMMER";I::
330 FOR C=1 TO 4
340 PRINT D$(C,I)
350 NEXT C
360 PRINT :D$(5,I)::::"WAHLLISTE
:"::"1- DATEN VERAENDERN":"2- DA
TEN AUSDRUCKEN":"3- BLAETTERN":"
4- WEITER"
370 CALL SOUND(150,1400,5)
380 CALL KEY(0,K,S)
390 IF S=0 THEN 380
400 IF K=49 THEN 450
410 IF K=50 THEN 560
420 IF K=51 THEN 670
430 IF K=52 THEN 20
440 GOTO 380
450 CALL CLEAR
460 PRINT "WELCHE ZEILE?(1-5)"
470 CALL SOUND(150,1400,5)
480 CALL KEY(0,K,S)
490 IF S=0 THEN 480
500 IF K<49 THEN 480
510 IF K>53 THEN 480
520 PRINT D$(K-48,I)
530 INPUT "KORRIGIERTE ZEILE=?
      ":D$(K-48,I)
540 CALL CLEAR
550 GOTO 320
560 CALL CLEAR
570 INPUT "STARTZEILE,ENDZEILE(1
-5)=?  ":X1,X2
580 OPEN #2:"TP",OUTPUT
590 PRINT #2:STR$(I):"**********
**********************"
600 FOR X=X1 TO X2
610 PRINT #2:D$(X,I)
620 NEXT X
630 PRINT #2:"******************
**************"
640 CLOSE #2
650 CALL CLEAR
660 GOTO 320
670 CALL CLEAR
680 I=I+1
690 IF I<=A THEN 320
700 I=1
710 GOTO 320
720 OPEN #1:"CS1",INTERNAL,OUTPU
T,FIXED
730 PRINT #1:A   ,
740 FOR D=1 TO A
750 FOR E=1 TO 5
760 PRINT #1:D$(E,D)
770 NEXT E
780 NEXT D
790 CLOSE #1
800 GOTO 20
810 PRINT "GEBEN SIE BIS ZU 5 ZE
ILEN   EIN.DRUECKEN SIE EINFACH
DIETASTE ENTER,WENN SIE EINE"
820 PRINT "LEERZEILE MOECHTEN.DA
NKE!":::
830 A=A+1
840 FOR Z=1 TO 5
850 PRINT "ZEILE";Z
860 INPUT D$(Z,A)
870 IF LEN(D$(Z,A))>28 THEN 860
880 NEXT Z
890 GOTO 20
900 END
```

Wenn man die Daten auch zu Papier bringen will, um so aus der gespeicherten Adresse einen Adreßaufkleber zu machen, benötigt man zusätzlich einen Drucker. Das Programm ist vollständig modular, die Bedienerführung erfolgt über zwei Menüs. Das Hauptmenü verzweigt zu den folgenden Funktionen:

Daten laden
Hiermit können die auf der Kassette gespeicherten Daten eingelesen werden. Auf dem Bildschirm erscheinen die Systeminstruktionen für den Kassettenbetrieb.

Daten suchen
Hier kann man nach einem Datensatz suchen. Als Suchbegriff ist eine komplette Zeile des gesuchten Datensatzes einzugeben, der gefundene Satz wird dann angezeigt. An dieser Stelle kann nun über ein Untermenü der Datensatz korrigiert oder gedruckt werden, man kann auch vorwärts »blättern«.

Daten speichern
Dieses Modul speichert die Kartei wieder auf Kassette ab .

Neue Daten eingeben
Für jede Karteikarte stehen fünf Zeilen zur Verfügung. Leerzeilen können einfach mit <ENTER> erzeugt werden. Zeilen, die länger als eine Bildschirmzeile sind, werden nicht angenommen.

Katalog
Mit diesem Kommando kann man die ganze Kartei durchblättern, wenn man sich an kein Schlüsselwort mehr erinnert. Das Modul zeigt den Satz, auf dem der Zeiger gerade steht und verzweigt dann zum Menü des Moduls »Daten suchen«; man kann also weiterblättern sowie den gerade angezeigten Satz korrigieren oder drucken.

(Torsten Niemietz)

viel. Er müßte deshalb jedem seiner drei Miterben zunächst einmal den, von diesen jeweils selbst geschätzten, Erbteil ausbezahlen, also an A 10000 Mark, an B 12500 Mark und an C 9000 Mark. Diese drei Beträge ergeben zusammen 31500 Mark. Die Differenz von diesem Betrag zu den 45000 Mark beträgt 13500 Mark. Dieser Differenzbetrag wird nun in vier Teile geteilt und jedem der vier Neffen 3375 Mark zusätzlich gutgeschrieben, so daß
A, der 10000 Mark erwartet hat, 13375 Mark bekommt,
B, der 12500 Mark erwartet hat, 15875 Mark bekommt,
C, der 9000 Mark erwartet hat, 12375 Mark bekommt und
D, anstatt 45000 Mark nur 41625 Mark an seine Miterben auszahlen müßte. Jeder der vier Erben würde in diesem Fall also wesentlich mehr bekommen, als ihm seiner eigenen Meinung nach zugestanden hätte. Keiner hätte folglich das Recht unzufrieden zu sein. Es wäre ihm ja freigestanden, seinen Anteil höher zu bewerten und damit noch gerechter berücksichtigt zu werden.

Der TRS-80 — ein elektronischer Salomon

Für ein noch komplizierteres Verteilungsverfahren, bei dem mehrere unteilbare Sachen unter mehrere Partner aufzuteilen sind, wie dies häufig bei Erbauseinandersetzungen der Fall ist, wurde das folgende Programm erstellt. Mit Hilfe dieses Programms lassen sich schwierige Verteilungsprobleme mühelos im Dialog mit dem Rechner lösen. Das Programm läuft auf dem TRS-80 Modell I und III und auf dem Video-Genie mit erweitertem Basic. Es läßt sich aber ohne große Schwierigkeiten an das Level-II-Basic anpassen, indem man anstelle der »INKEY«-Abfragen und der »INSTR«-Statements, normale »INPUT«-Eingaben verwendet.

Nun aber zur Programmbeschreibung:

Die Zeilen 240 bis 280 enthalten, neben der Initialisierung, eine kurze Erläuterung und die Abfragen nach dem Thema, Anzahl der Beteiligten, Anzahl der Gegenstände und dem zur Verteilung kommenden Geldbetrag. Schließlich wird gefragt, ob eine Hardcopy gewünscht wird. In den Zeilen 320 bis 330 erfolgt die Abfrage nach den Namen der Beteiligten. Unabhängig von der Reihenfolge, in der die Eingabe erfolgt, werden die Namen in einer zufälligen Folge angeordnet, damit sich für den Schätzvorgang keiner einen Platzvorteil verschaffen kann. In Zeile 370 erfolgt die Eingabe der zur Verteilung anstehenden Sachen.

In den Zeilen 410 bis 430 wird der Schätzvorgang erläutert.

In Zeile 470 werden die Schätzwerte eingegeben.

Die Zeilen 510 bis 640 enthalten das Listprogramm für die Schätzwertliste.

Die Zeilen 680 bis 700 enthalten den Programmteil für die Auflistung aus der hervorgeht, wem was bei der Verteilung zufällt.

In den Zeilen 680 bis 700 wird die Liste für den finanziellen Ausgleich erzeugt.

Die Zeilen 740 bis 810 enthalten die verschiedenen Möglichkeiten zur Programmbeendigung.

Die Zeile 850 enthält eine Unterroutine zum Ausdruck der Namenszeile für jede einzelne Liste.

Die Zeilen 910 bis 920 drucken die Überschriften zu den Hardcopien der Bildschirmausdrucke.

Die Zeilen 960 bis 980 enthalten eine Routine zum Ausdrucken des Bildschirminhaltes auf dem Schnelldrucker.

Mit dem Bildschirmausdruck eines mit diesem Programm errechneten Verteilungsschlüssels wird die Funktionsweise des Programms demonstriert. Bei diesem Programm geht es weniger darum, ein besonders elegantes Programmierbeispiel zu geben, sondern vielmehr darum, zu zeigen, was man mit einem Computer alles machen kann und dabei ein etwas ungewöhnliches Beispiel vorzustellen, das für viele mit einem verblüffenden Ergebnis endet. (Josef Menz)

Der TI 99/4A als Mathematiklehrer

Wohl kaum ein Lehrer hat soviel Geduld im Nachhilfestunden geben, wie der TI 99/4A. Sollte speziell Mathematik Schwierigkeiten bereiten, so kann das nachfolgende Programm viele Nachhilfestunden ersparen.

Das Mathematik-Übungsprogramm für den TI 99/4A in Extended Basic erlaubt das Üben der vier Grundrechenarten sowie des Quadrierens und des Radizierens. Nach Auswahl der Rechenart wird vom Programm gefragt, in welchem Zahlenbereich die Übungsaufgaben liegen müssen und wieviel Aufgaben der Computer stellen soll. Anschließend werden die Aufgaben gestellt und die Lösungen vom Benutzer angefordert. Ist die Lösung richtig, so wird der Bildschirm grün — ist die Lösung falsch, so wird gefragt, ob man die Aufgabe noch einmal vesuchen will oder ob das Ergebnis bekanntgegeben werden soll. Nach der letzten Aufgabe wird mitgeteilt, wieviele Aufgaben falsch gelöst wurden und welche Note man für das Ergebnis erhält. Das Programm »Mathematik Übungen« hat eine Länge von zirka 6,2 KByte.

(Gerhard Külkens)

```
100 REM +----------------+
        + COPYRIGHT BY: +
        + GERH.KUELKENS +
        + EXT.-BASIC TI +
        +----------------+
110 REM +----------------+
        + AUGUST  1983  +
        + VERSION 2.1   +
        + TI 99/4A      +
        +----------------+
120 CALL CLEAR
130 CALL SCREEN(16)
140 DISPLAY AT(1,1):"*** MATHEMA
    TIK-UEBUNGEN ***"
150 DISPLAY AT(5,2):"BITTE WAEHL
    EN SIE AUS:"
160 DISPLAY AT(8,3):"ADDIEREN...
    .......=1"
170 DISPLAY AT(10,3):"SUBTRAHIER
    EN.....=2"
180 DISPLAY AT(12,3):"MULTIPLIZI
    EREN....=3"
190 DISPLAY AT(14,3):"DIVIDIEREN
    ........=4"
200 DISPLAY AT(16,3):"QUADRIEREN
    ........=5"
210 DISPLAY AT(18,3):"RADIZIEREN
    ........=6"
220 DISPLAY AT(20,3):"PROGRAMMEN
    DE......=7"
230 DISPLAY AT(23,2)BEEP:"WELCHE
     RECHENART ? "
240 CALL KEY(0,R,S)
250 IF S=0 THEN 240
260 R=(R-48)
270 IF R<1 THEN 240 ELSE 280
280 IF R>7 THEN 300
290 IF R>7 THEN 240 ELSE 330
300 CALL CLEAR
310 END
320 REM + AUFGABEN-UMFANG +
330 CALL CLEAR
340 CALL SCREEN(12)
350 DISPLAY AT(2,1):"**** AUFG
    ABENUMFANG ****"
360 DISPLAY AT(5,1):"WIE KLEIN S
    OLL DIE WIEDRIG-"
370 DISPLAY AT(7,1):"STE UND WIE
     HOCH SOLL DIE"
380 DISPLAY AT(9,1):"HOECHSTE ZA
    HL SEIN ?."
390 DISPLAY AT(13,2):"KLEINSTE ?
    "
400 DISPLAY AT(15,2):"GROESSTE ?
```

Listing zu »Mathematik-Übungen«

```
410 DISPLAY AT(19,1):"WIEVIELE A
    UFGABEN SOLL ICH"
420 DISPLAY AT(21,1):"STELLEN ?.
    "
430 DISPLAY AT(24,2):"ANZAHL ?."
440 ACCEPT AT(13,13)BEEP:E
450 ACCEPT AT(15,13)BEEP:B
460 ACCEPT AT(24,12)BEEP:W
470 ON R GOTO 490,660,840,1010,1
    190,1380
480 REM +----------------+
        +   ADDITION    +
        +----------------+
490 FOR I=1 TO W STEP 1
500 CALL CLEAR
510 CALL SCREEN(16)
520 DISPLAY AT(1,1):"****** AD
    DITIONEN ******"
530 DISPLAY AT(5,2):"AUFGABE NR.
    ";I
540 IF XX=1 THEN 560
550 GOSUB 1590
560 DISPLAY AT(8,2):Z1;"+";Z2;"=
    ?"
570 E1=Z1+Z2
580 DISPLAY AT(11,2):"IHR ERGEBN
    IS ? "
590 ACCEPT AT(11,19)BEEP:E
600 GOSUB 1660
610 IF XX=1 THEN 500 ELSE 620
620 IF I=W THEN 640 ELSE 630
630 NEXT I
640 GOSUB 2000
650 REM +----------------+
        +  SUBTRAKTION  +
        +----------------+
660 FOR I=1 TO W STEP 1
670 CALL CLEAR
680 CALL SCREEN(16)
690 DISPLAY AT(1,1):"***** SUBT
    RAKTIONEN *****"
700 DISPLAY AT(5,2):"AUFGABE NR.
    ";I
710 IF XX=1 THEN 740
720 GOSUB 1590
730 IF Z1<Z2 THEN 720 ELSE 740
740 DISPLAY AT(8,2):Z1;"-";Z2;"=
    ?"
750 E1=Z1-Z2
760 DISPLAY AT(11,2):"IHR ERGEBN
    IS ? "
770 ACCEPT AT(11,19)BEEP:E
780 GOSUB 1660
790 IF XX=1 THEN 670 ELSE 800
800 IF I=W THEN 820 ELSE 810
810 NEXT I
820 GOSUB 2000
830 REM +----------------+
        + MULTIPLIKATION +
        +----------------+
840 FOR I=1 TO W STEP 1
850 CALL CLEAR
860 CALL SCREEN(16)
870 DISPLAY AT(1,1):" **** MULTI
    PLIKATIONEN ****"
880 DISPLAY AT(5,2):"AUFGABE NR.
    ";I
890 IF XX=1 THEN 910
900 GOSUB 1590
910 DISPLAY AT(8,2):Z1;"*";Z2;"=
    ?"
920 E1=Z1*Z2
930 DISPLAY AT(11,2):"IHR ERGEBN
    IS ? "
940 ACCEPT AT(11,19)BEEP:E
950 GOSUB 1660
960 IF XX=1 THEN 850 ELSE 970
970 IF I=W THEN 990 ELSE 980
980 NEXT I
990 GOSUB 2000
1000 REM +----------------+
         +   DIVISION    +
         +----------------+
1010 FOR I=1 TO W STEP 1
1020 CALL CLEAR
1030 CALL SCREEN(16)
1040 DISPLAY AT(1,1):" ****** D
     IVISIONEN ******"
1050 DISPLAY AT(5,2):"AUFGABE NR
     .";I
1060 IF XX=1 THEN 1100
1070 GOSUB 1590
1080 IF Z1=0 THEN 1070
1090 Z1=INT(Z1*Z2)
1100 DISPLAY AT(8,2):Z1;":";Z2;"
     = ?"
1110 E1=Z1/Z2
1120 DISPLAY AT(11,2):"IHR ERGEB
     NIS ?"
1130 ACCEPT AT(11,19)BEEP:E
1140 GOSUB 1660
1150 IF XX=1 THEN 1020 ELSE 1160
1160 IF I=W THEN 1180 ELSE 1170
1170 NEXT I
1180 GOSUB 2000
1190 REM +----------------+
         +  QUADRIERUNG  +
         +----------------+
1200 FOR I=1 TO W STEP 1
1210 CALL CLEAR
1220 CALL SCREEN(16)
1230 DISPLAY AT(1,1):" ****** Q
     UADRIEREN ******"
1240 DISPLAY AT(5,2):"AUFGABE NR
     .";I
1250 IF XX=1 THEN 1290
1260 GOSUB 1590
1270 IF Z1=0 THEN 1260
1280 IF Z1>6 THEN 1260
1290 DISPLAY AT(8,2):"QUADRIERE
     ";Z1;"EXP 2";"= ?"
1300 E1=Z1^2
1310 DISPLAY AT(11,2):"IHR ERGEB
     NIS ?"
1320 ACCEPT AT(11,19)BEEP:E
1330 GOSUB 1660
1340 IF XX=1 THEN 1210 ELSE 1160
1350 IF I=W THEN 1370 ELSE 1360
1360 NEXT I
1370 GOSUB 2000
1380 REM +----------------+
         +  RADIZIERUNG  +
         +----------------+
1390 FOR I=1 TO W STEP 1
1400 CALL CLEAR
1410 CALL SCREEN(16)
1420 DISPLAY AT(1,1):" ****** R
     ADIZIEREN ******"
1430 DISPLAY AT(5,2):"AUFGABE NR
     .";I
1440 IF XX=1 THEN 1480
1450 GOSUB 1590
1460 IF Z1<2 THEN 1450
1470 Z1=INT(Z1*Z1)
1480 DISPLAY AT(8,2):"WURZEL AUS
     ";Z1;"= ?"
1490 E1=SQR(Z1)
1500 DISPLAY AT(11,2):"IHR ERGEB
     NIS ?"
1510 ACCEPT AT(11,19)BEEP:E
1520 GOSUB 1660
1530 IF XX=1 THEN 1400 ELSE 1160
1540 IF I=W THEN 1560 ELSE 1550
1550 NEXT I
1560 GOSUB 2000
1570 REM
1580 REM
1590 REM +----------------+
         + ZUFALLSZAHLEN +
         +----------------+
1600 RANDOMIZE
1610 Z1=INT(RND*(B-K))+K
1620 IF SGN(Z1)=-1 THEN 1610
1630 Z2=INT(RND*(B-K))+K
1640 IF SGN(Z2)=-1 THEN 1630
1650 RETURN
1660 REM +----------------+
         + ERGEBNISPRUEF. +
         +----------------+
1670 IF E<>E1 THEN 1680 ELSE 186
     0
1680 N=N+1
1690 CALL SCREEN(10)
1700 CALL SOUND(100,110,1)
1710 DISPLAY AT(15,1):"** PARDON
      DAS WAR FALSCH! **"
1720 DISPLAY AT(17,1):"WIEDERHOL
     EN,-ODER SOLL ICH"
1730 DISPLAY AT(19,1):"DAS ERGEB
     NIS BEKANNT GEBEN ?"
1740 DISPLAY AT(21,1)BEEP:"JA-J,
     NEIN=N "
1750 CALL KEY(2,EB,S)
1760 IF S=0 THEN 1750
1770 IF EB=15 THEN 1800
1780 IF EB=2 THEN 1820
1790 GOTO 1750
1800 XX=1
1810 RETURN
1820 DISPLAY AT(19,1):"DAS ERGEB
     NIS LAUTET :"
1830 XX=0
1840 ON R GOSUB 2230,2250,2270,2
     290,2310,2330
1850 GOTO 1960
1860 CALL SOUND(100,440,2)
1870 CALL SOUND(100,550,2)
1880 CALL SOUND(100,440,2)
1890 CALL SOUND(100,660,2)
1900 CALL SOUND(100,880,2)
1910 CALL SCREEN(4)
1920 DISPLAY AT(15,1):"**********
     ********************"
1930 DISPLAY AT(17,1):"GRATULIER
     E DAS WAR RICHTIG !"
1940 DISPLAY AT(19,1):"**********
     ********************"
1950 XX=0
1960 DISPLAY AT(24,1)BEEP:"WEITE
     R MIT BELIEBIGER TASTE!"
1970 CALL KEY(0,K,S)
1980 IF S=0 THEN 1970
1990 RETURN
2000 REM +----------------+
         + DURCHSCHNITTS- +
         + BERECHNUNGEN  +
         +----------------+
2010 IF N=0 THEN 2020 ELSE 2040
2020 PR=0
2030 GOTO 2050
2040 PR=INT(N/W*100)
2050 CALL CLEAR
2060 CALL SCREEN(12)
2070 DISPLAY AT(2,1):"VON";W;"DE
     STELLTEN AUFGABEN"
2080 DISPLAY AT(4,1):"WAREN";N;"
     FALSCH"
2090 DISPLAY AT(8,1):"DAS ENTSPR
     ICHT EINER FEHLER"
2100 DISPLAY AT(10,1):"QUOTE VON
     ";PR;"%"
2110 DISPLAY AT(13,1):"IHRE NOTE
     IST:"
2120 IF PR<9 THEN 2350
2130 IF PR<18 THEN 2370
2140 IF PR<34 THEN 2390
2150 IF PR<49 THEN 2410
2160 IF PR<64 THEN 2430
2170 IF PR>65 THEN 2450
2180 DISPLAY AT(24,1):"WEITER MI
     T BELIEBIGER TASTE!"
2190 N=0
2200 CALL KEY(0,K,S)
2210 IF S=0 THEN 2200
2220 GOTO 120
2230 DISPLAY AT(21,1):Z1;" +";Z2
     ;"=";E1
2240 RETURN
2250 DISPLAY AT(21,1):Z1;" -";Z2
     ;"=";E1
2260 RETURN
2270 DISPLAY AT(21,1):Z1;" *";Z2
     ;"=";E1
2280 RETURN
2290 DISPLAY AT(21,1):Z1;" :";Z2
     ;"=";E1
2300 RETURN
2310 DISPLAY AT(21,1):Z1;" EXP.2
     ";"=";E1
2320 RETURN
2330 DISPLAY AT(21,1):"WURZEL AU
     S ";Z1;"=";E1
2340 RETURN
2350 DISPLAY AT(18,1):"******  1
     =SEHR GUT  ******"
2360 GOTO 2180
2370 DISPLAY AT(18,1):"*****
       2=GUT      *****"
2380 GOTO 2180
2390 DISPLAY AT(18,1):"****  3=B
     EFRIEDIGEND  ****";
2400 GOTO 2180
2410 DISPLAY AT(18,1):"***    4=
     AUSREICHEND    ***"
2420 GOTO 2180
2430 DISPLAY AT(18,1):"----    5
     =MANGELHAFT    ----"
2440 GOTO 2180
2450 DISPLAY AT(18,1):"------ 6=
     UNGENUEGEND ------"
2460 GOTO 2180
2470 !DIESES PROGRAMM DIENT
      ZUR UEBUNG.EINE ER-
       KLAERUNG IST NICHT ER-
        FORDERLICH DA ALLE EIN-
         ZELNEN EINGABEN IM
2480 !DIALOGBETRIEB ABGEAR-
      BEITET WERDEN.
2490 !DAS PROGRAMM BENOETIGT
      6678 BYTES AN SPEICHER-
       PLATZ.
2500 !UEBERSICHT:
2510 !AUTOM.AUFGABENSTELLUNG
2520 !VAR. ZAHLENBEREICH
2530 !KORR. VON AUFGABEN
2540 !GRAPHISCHE UND
2550 !AKUSTISCHE WIEDERGABE
2560 !ZENSURENVERGABE
```

Listing zu »Mathematik-Übungen« (Schluß)

Bibliotheksverwaltung

Diese »Bibliotheksverwaltung« für den TI 99/4 ermöglicht es, Daten von Büchern einer Bibliothek zu erfassen und bei Bedarf Bücher nach bestimmten Gesichtspunkten herauszusuchen.

TI 99/4

Eine Hauptauswahlliste (Bild 1) verzweigt in drei Module: Modul 1 (ab Zeile 10000) ermöglicht das Einlesen von Datensätzen vom Kassettenrecorder. Modul 2 (ab Zeile 1000) verzweigt in ein Menü. Modul 3 (ab Zeile 500) ermöglicht den Ausstieg aus dem Programm. In Modul 2 kann man folgende Untermodule anwählen:

□ Bucheingabe (ab Zeile 2000)

Nach Eingabe der Daten eines Buches wird abgefragt, ob eine weitere Eingabe erwünscht ist. Ist dies der Fall, wird die Anzahl der Sätze automatisch um 1 erhöht und eine weitere »Bucheingabe« kann folgen. Wird dies nicht gewünscht, erfolgt der Rücksprung ins Menü.

Vom Modul »Bucheingabe« wird ein Unterprogramm angesprungen, welches die »INPUT-Statements« enthält (ab Zeile 12000). Die Feldnamen in diesen Statements (Autor, Titel und so weiter) können natürlich nach eigenem Ermessen abgeändert werden. Außerdem wird in diesem Unterprogramm die Art eines jeden Feldes festgelegt: ob es alphanumerische Daten (FA = 1) oder nur numerische Daten (FA = 2) ent-

```
1 REM  **********************
2 REM  BIBLIOTHEKSVERWALTUNG
3 REM  **********************
4 REM  PROGRAMMAUTOREN:
5 REM  ROLAND BEER
6 REM  WERNER LOSERT
7 REM  RUEDIGER VASA
8 REM  **********************
9 REM  NOVEMBER 1982
10 REM
11 REM
12 DIM SF$(30,10)
20 DIM FA(10)
30 DN$="BIBLIOTHEKSVERWALTUNG"
100 CALL CLEAR
101 BF=5
102 HF=15
103 GOSUB 15000
110 PRINT TAB(5);"HAUPTWAHLLISTE
":::::::TAB(5);"1 - BESTAND EINLES
EN":;TAB(5);"2 - MENUE"
120 PRINT TAB(5);"3 - ENDE":::::
TAB(5);"BITTE WAEHLEN SIE!":::::
::
130 CALL VCHAR(6,3,42,14)
140 CALL VCHAR(6,30,42,14)
150 CALL HCHAR(5,3,42,28)
160 CALL HCHAR(20,3,42,28)
161 CALL HCHAR(9,3,42,28)
162 CALL HCHAR(16,3,42,28)
163 CALL VCHAR(6,5,42,14)
164 CALL VCHAR(6,28,42,14)
165 CALL VCHAR(6,4,63,14)
166 CALL VCHAR(6,29,63,14)
170 CALL KEY(0,T,Z)
180 IF Z=0 THEN 170
190 IF T=49 THEN 10000
200 IF T=50 THEN 1000
210 IF T=51 THEN 500
230 GOTO 170
500 CALL CLEAR
501 PRINT ::::::::::::::::::"UM D
AS PROGRAMM NEU":"ZU STARTEN, BI
TTE"::"  - RUN -  "::"EINGEBEN !
"::::::
510 END
1000 CALL CLEAR
1001 BF=7
1002 HF=12
1003 GOSUB 15000
1010 PRINT TAB(10);"M E N U E"::
::TAB(4);"1 - BUCHEINGABE":;TAB
(4);"2 - BILDSCHIRMAUSGABE"
1020 PRINT TAB(4);"3 - KATALOG":
;TAB(4);"4 - HAUPTWAHLLISTE":;TA
B(4);"5 - SPEICHERN"
1021 PRINT ::
1030 PRINT TAB(6);"BITTE WAEHLEN
 SIE!":::::::
1040 CALL VCHAR(5,3,42,15)
1050 CALL VCHAR(5,30,42,15)
1060 CALL HCHAR(5,3,42,28)
1070 CALL HCHAR(20,3,42,28)
1071 CALL HCHAR(9,3,42,28)
1072 CALL HCHAR(16,3,42,28)
1073 CALL VCHAR(5,5,42,15)
1074 CALL VCHAR(5,28,42,15)
1075 CALL VCHAR(6,4,63,14)
1076 CALL VCHAR(6,29,63,14)
1080 CALL KEY(0,T,Z)
1090 IF Z=0 THEN 1080
1100 IF T=49 THEN 2000
1110 IF T=50 THEN 3000
1120 IF T=51 THEN 4000
1130 IF T=52 THEN 100
1135 IF T=53 THEN 8000
1140 GOTO 1080
2000 REM
2001 BF=10
2002 HF=15
2003 GOSUB 15000
2010 I=SZ+1
2020 GOSUB 12000
2030 SZ=I
2040 GOSUB 14000
2045 PRINT ::"WEITERE BUCHEINGAB
EN ? (J/N)  "
2046 CALL KEY(0,T,Z)
2047 IF Z=0 THEN 2046
2048 IF T=74 THEN 2010
2049 IF T=78 THEN 1000
2050 GOTO 2046
3000 CALL CLEAR
3001 BF=13
3002 HF=4
3003 GOSUB 15000
3010 PRINT TAB(3);"AUSGABE AUF B
ILDSCHIRM":::::TAB(3);"1 - BUCHN
UMMER BEKANNT ?"
3015 PRINT TAB(3);"2 - BUCHNUMME
R UNBEKANNT ?":::::
3020 PRINT TAB(3);"WAEHLEN SIE 1
 ODER 2!"
3030 CALL KEY(0,T,Z)
3040 IF Z=0 THEN 3030
3050 IF T=49 THEN 3100
```

Bild 1. Listing des Programms zur Verwaltung einer Bibliothek

Bibliotheksverwaltung

Diese »Bibliotheksverwaltung« für den TI 99/4 ermöglicht es, Daten von Büchern einer Bibliothek zu erfassen und bei Bedarf Bücher nach bestimmten Gesichtspunkten herauszusuchen.

TI 99/4

Eine Hauptauswahlliste (Bild 1) verzweigt in drei Module: Modul 1 (ab Zeile 10000) ermöglicht das Einlesen von Datensätzen vom Kassettenrecorder. Modul 2 (ab Zeile 1000) verzweigt in ein Menü. Modul 3 (ab Zeile 500) ermöglicht den Ausstieg aus dem Programm. In Modul 2 kann man folgende Untermodule anwählen:

□ Bucheingabe (ab Zeile 2000)

Nach Eingabe der Daten eines Buches wird abgefragt, ob eine weitere Eingabe erwünscht ist. Ist dies der Fall, wird die Anzahl der Sätze automatisch um 1 erhöht und eine weitere »Bucheingabe« kann folgen. Wird dies nicht gewünscht, erfolgt der Rücksprung ins Menü.

Vom Modul »Bucheingabe« wird ein Unterprogramm angesprungen, welches die »INPUT-Statements« enthält (ab Zeile 12000). Die Feldnamen in diesen Statements (Autor, Titel und so weiter) können natürlich nach eigenem Ermessen abgeändert werden. Außerdem wird in diesem Unterprogramm die Art eines jeden Feldes festgelegt: ob es alphanumerische Daten (FA = 1) oder nur numerische Daten (FA = 2) ent-

```
1 REM  **********************
2 REM  BIBLIOTHEKSVERWALTUNG
3 REM  **********************
4 REM  PROGRAMMAUTOREN:
5 REM  ROLAND BEER
6 REM  WERNER LOSERT
7 REM  RUEDIGER VASA
8 REM  **********************
9 REM  NOVEMBER 1982
10 REM
11 REM
12 DIM SF$(30,10)
20 DIM FA(10)
30 DN$="BIBLIOTHEKSVERWALTUNG"
100 CALL CLEAR
101 BF=5
102 HF=15
103 GOSUB 15000
110 PRINT TAB(5);"HAUPTWAHLLISTE
":::::::TAB(5);"1 - BESTAND EINLES
EN":;TAB(5);"2 - MENUE"
120 PRINT TAB(5);"3 - ENDE":::::
TAB(5);"BITTE WAEHLEN SIE!":::::
::
130 CALL VCHAR(6,3,42,14)
140 CALL VCHAR(6,30,42,14)
150 CALL HCHAR(5,3,42,28)
160 CALL HCHAR(20,3,42,28)
161 CALL HCHAR(9,3,42,28)
162 CALL HCHAR(16,3,42,28)
163 CALL VCHAR(6,5,42,14)
164 CALL VCHAR(6,28,42,14)
165 CALL VCHAR(6,4,63,14)
166 CALL VCHAR(6,29,63,14)
170 CALL KEY(0,T,Z)
180 IF Z=0 THEN 170
190 IF T=49 THEN 10000
200 IF T=50 THEN 1000
210 IF T=51 THEN 500
230 GOTO 170
500 CALL CLEAR
501 PRINT ::::::::::::::::::"UM D
AS PROGRAMM NEU":"ZU STARTEN, BI
TTE"::"  - RUN -  "::"EINGEBEN !
"::::::
510 END
1000 CALL CLEAR
1001 BF=7
1002 HF=12
1003 GOSUB 15000
1010 PRINT TAB(10);"M E N U E"::
::TAB(4);"1 - BUCHEINGABE":;TAB
(4);"2 - BILDSCHIRMAUSGABE"
1020 PRINT TAB(4);"3 - KATALOG":
;TAB(4);"4 - HAUPTWAHLLISTE":;TA
B(4);"5 - SPEICHERN"
1021 PRINT ::
1030 PRINT TAB(6);"BITTE WAEHLEN
 SIE!":::::::
1040 CALL VCHAR(5,3,42,15)
1050 CALL VCHAR(5,30,42,15)
1060 CALL HCHAR(5,3,42,28)
1070 CALL HCHAR(20,3,42,28)
1071 CALL HCHAR(9,3,42,28)
1072 CALL HCHAR(16,3,42,28)
1073 CALL VCHAR(5,5,42,15)
1074 CALL VCHAR(5,28,42,15)
1075 CALL VCHAR(6,4,63,14)
1076 CALL VCHAR(6,29,63,14)
1080 CALL KEY(0,T,Z)
1090 IF Z=0 THEN 1080
1100 IF T=49 THEN 2000
1110 IF T=50 THEN 3000
1120 IF T=51 THEN 4000
1130 IF T=52 THEN 100
1135 IF T=53 THEN 8000
1140 GOTO 1080
2000 REM
2001 BF=10
2002 HF=15
2003 GOSUB 15000
2010 I=SZ+1
2020 GOSUB 12000
2030 SZ=I
2040 GOSUB 14000
2045 PRINT ::"WEITERE BUCHEINGAB
EN ? (J/N)  "
2046 CALL KEY(0,T,Z)
2047 IF Z=0 THEN 2046
2048 IF T=74 THEN 2010
2049 IF T=78 THEN 1000
2050 GOTO 2046
3000 CALL CLEAR
3001 BF=13
3002 HF=4
3003 GOSUB 15000
3010 PRINT TAB(3);"AUSGABE AUF B
ILDSCHIRM":::::TAB(3);"1 - BUCHN
UMMER BEKANNT ?"
3015 PRINT TAB(3);"2 - BUCHNUMME
R UNBEKANNT ?":::::
3020 PRINT TAB(3);"WAEHLEN SIE 1
 ODER 2!"
3030 CALL KEY(0,T,Z)
3040 IF Z=0 THEN 3030
3050 IF T=49 THEN 3100
```

Bild 1. Listing des Programms zur Verwaltung einer Bibliothek

```
3060 IF T=50 THEN 3500
3070 GOTO 3030
3100 CALL CLEAR
3110 PRINT ::::::::::::::::::::::::
:::"GEBEN SIE BITTE DIE"
3120 INPUT "BUCHNUMMER EIN:  ":I
3130 CALL CLEAR
3131 BF=15
3132 HF=6
3133 GOSUB 15000
3140 PRINT ::::::::::::::::::::::::
:::"BUCHNUMMER:  ";I::
3150 FOR K=1 TO 6
3160 PRINT SF$(I,K)
3170 NEXT K
3180 PRINT ::"**********************
**********"
3190 PRINT "WAHLLISTE":"1 - VORW
AERTS BLAETTERN":"2 - RUECKWAERT
S BLAETTERN":"3 - DRUCKEN"
3200 PRINT "4 - AENDERUNGEN":"5
- MENUE":"**********************
******"::
3210 CALL KEY(0,T,Z)
3220 IF Z=0 THEN 3210
3230 IF T=49 THEN 3290
3240 IF T=50 THEN 3330
3250 IF T=51 THEN 6000
3260 IF T=52 THEN 7000
3270 IF T=53 THEN 1000
3280 GOTO 3210
3290 I=I+1
3300 IF I>SZ THEN 3310 ELSE 3130
3310 I=1
3320 GOTO 3130
3330 I=I-1
3340 IF I<1 THEN 3350 ELSE 3130
3350 I=SZ
3360 GOTO 3130
3500 CALL CLEAR
3501 BF=15
3502 HF=6
3503 GOSUB 15000
3510 PRINT ::::::::::::::::::::::::
:::"GEBEN SIE 2 STICHWORTE EIN!"
:::
3520 INPUT "STICHWORT 1: ":SUCH1
$
3530 INPUT "STICHWORT 2: ":SUCH2
$
3540 M=1
3550 FOR I=M TO SZ
3560 FOR K=1 TO 10
3570 IF SUCH1$=SF$(I,K)THEN 3610
3580 NEXT K
3590 NEXT I
3600 GOTO 3660
3610 FOR K=1 TO 10
3620 IF SUCH2$=SF$(I,K)THEN 3700
3630 NEXT K
3640 M=M+1
3650 IF M>SZ THEN 3660 ELSE 3550
3660 PRINT ::SUCH1$;" UND ";SUCH
2$:"SIND NICHT ZU FINDEN!"
3670 FOR V=1 TO 500
3680 NEXT V
3690 GOTO 1000
3700 CALL CLEAR
3709 PRINT ::::::::::::::::::::::::
:::
3710 FOR K=1 TO 6
3720 PRINT SF$(I,K)
3730 NEXT K
3740 PRINT ::::"1 - GEFUNDENES B
UCH IST":"GESUCHTES BUCH.":"2 -
GEFUNDENES BUCH IST":"      N I
C H T"
3745 PRINT "GESUCHTES BUCH."
3750 PRINT "WAEHLEN SIE 1 ODER 2
!"
3760 CALL KEY(0,T,Z)
3770 IF Z=0 THEN 3760
3780 IF T=49 THEN 3130
3790 IF T=50 THEN 3810
3800 GOTO 3760
3810 M=M+1
3815 IF M>SZ THEN 3660 ELSE 3550
4000 CALL CLEAR
4001 BF=13
4002 HF=12
4003 GOSUB 15000
4010 PRINT "K A T A L O G":::
4020 FOR I=1 TO SZ
4030 PRINT SF$(I,1);" *  BUCHNUM
MER:  ";I
4040 NEXT I
4050 PRINT ::"1 - RUECKSPRUNG IN
S MENUE"
4060 CALL KEY(0,T,Z)
4070 IF Z=0 THEN 4060
4080 IF T=49 THEN 1000 ELSE 4060
6000 CALL CLEAR
6001 BF=9
6002 HF=16
6003 GOSUB 15000
6010 PRINT "DRUCKER ANGESCHLOSSE
N ???"::"WENN JA, BITTE -ENTER-
DRUECKEN!":::"RUECKSPRUNG IN DIE
 AUSGABE"
6020 PRINT "DRUECKEN SIE IRGENDE
INE TASTE."
6030 CALL KEY(0,T,Z)
6040 IF Z=0 THEN 6030
6050 IF T<>13 THEN 3130
6060 OPEN #1:"TP",OUTPUT
6070 PRINT #1:"BUCHNUMMER:  ";I
6080 FOR K=1 TO 6
6090 PRINT #1:SF$(I,K)
6100 NEXT K
6110 CLOSE #1
6120 GOTO 3130
7000 CALL CLEAR
7001 BF=8
7002 HF=6
7003 GOSUB 15000
7010 PRINT "BUCHNUMMER:  ";I
7020 FOR K=1 TO 10
7030 PRINT K;")  ";SF$(I,K)
7040 NEXT K
7050 PRINT ::
7060 INPUT "WELCHES FELD AENDERN
? ":K
7070 INPUT "NEUER FELDINHALT: ":
SF$(I,K)
7080 PRINT ::"1 - AENDERUNGEN AB
GESCHLOSSEN?":"2 - WEITERE AENDE
RUNGEN?"
7090 CALL KEY(0,T,Z)
7100 IF Z=0 THEN 7090
7110 IF T=49 THEN 3130
7120 IF T=50 THEN 7000
7130 GOTO 7090
8000 CALL CLEAR
8010 CALL SCREEN(11)
8020 OPEN #2:"CS1",OUTPUT,INTERN
AL,FIXED 192
8030 PRINT #2:DN$,SZ
8040 FOR I=1 TO SZ
8050 FOR K=1 TO 10
8060 PRINT #2:SF$(I,K)
8070 NEXT K
8080 NEXT I
8090 CLOSE #2
8100 GOTO 1000
10000 CALL CLEAR
10001 CALL SCREEN(16)
10005 OPEN #3:"CS1",INPUT ,INTER
NAL,FIXED 192
10010 INPUT #3:DN$,SZ
10020 FOR I=1 TO SZ
```

Bild 1. Listing des Programms zur Verwaltung einer Bibliothek (Fortsetzung)

halten soll. Ein gesondertes Unterprogramm ab Zeile 13000 führt eine Eingabeüberprüfung durch und gibt bei fehlerhafter Eingabe eine entsprechende Meldung aus, so daß die Eingabe wiederholt werden kann.

Jedes Buch wird automatisch nach Autoren alphabetisch eingeordnet. Dazu wird in ein Unterprogramm »Sortieren« (ab Zeile 14000) gesprungen.

□ Ausgabe auf dem Bildschirm (ab Zeile 3000)

Bei diesem Modul ergeben sich zwei Möglichkeiten. Die erste ist die, daß man die Buchnummer eingibt, falls diese bekannt ist. Die zweite Möglichkeit aktiviert ein Suchprogramm (ab Zeile 3500). Dabei müssen

```
10030 FOR K=1 TO 10
10040 INPUT #3:SF$(I,K)
10050 NEXT K
10060 NEXT I
10070 CLOSE #3
10080 GOTO 100
12000 REM
12010 REM
12020 REM   UP EINGABE
12030 REM
12040 REM
12050 CALL CLEAR
12060 PRINT ::::::::::::::::::::
::::"BUCHNUMMER ";I::
12070 K=1
12080 INPUT "AUTOR       :":SF$(I
,K)
12090 FA(K)=1
12100 GOSUB 13000
12110 IF FEHLER=1 THEN 12080
12120 K=2
12130 INPUT "TITEL       :":SF$(I
,K)
12140 FA(K)=1
12150 GOSUB 13000
12160 IF FEHLER=1 THEN 12130
12170 K=3
12180 INPUT "ERSCH.JAHR :":SF$(I
,K)
12190 FA(K)=1
12200 GOSUB 13000
12210 IF FEHLER=1 THEN 12180
12220 K=4
12230 INPUT "ERSCH.ORT   :":SF$(I
,K)
12240 FA(K)=1
12250 GOSUB 13000
12260 IF FEHLER=1 THEN 12230
12270 K=5
12280 INPUT "BESTAND     :":SF$(I
,K)
12290 FA(K)=2
12300 GOSUB 13000
12310 IF FEHLER=1 THEN 12280
12320 K=6
12330 INPUT "ABLAGE.NR. :":SF$(I
,K)
12340 FA(K)=1
12350 GOSUB 13000
12360 IF FEHLER=1 THEN 12330
12370 K=7
12380 INPUT "STICHWORT 1:":SF$(I
,K)
12390 FA(K)=1
12400 GOSUB 13000
12410 IF FEHLER=1 THEN 12380
12420 K=8
12430 INPUT "STICHWORT 2:":SF$(I
,K)
12440 FA(K)=1
12450 GOSUB 13000
12460 IF FEHLER=1 THEN 12430
12470 K=9
12480 INPUT "STICHWORT 3:":SF$(I
,K)
12490 FA(K)=1
12500 GOSUB 13000
12510 IF FEHLER=1 THEN 12480
12520 K=10
12530 INPUT "STICHWORT 4:":SF$(I
,K)
12540 FA(K)=1
12550 GOSUB 13000
12560 IF FEHLER=1 THEN 12530
12570 RETURN
13000 REM
13010 REM
13020 REM   UP PRUEF
13030 REM
13040 REM
13050 PRUEF$=SF$(I,K)
13060 ART=FA(K)
13070 FEHLER=0
13080 IF ART=1 THEN 13310
13090 LAENGE=LEN(PRUEF$)
13100 IF LAENGE=0 THEN 13290
13110 MINUS=0
13120 PUNKT=0
13130 ZIFFER=0
13140 FOR L=1 TO LAENGE
13150 ASCII=ASC(SEG$(PRUEF$,L,1)
)
13160 'IF ASCII<>46 THEN 13180
13170 PUNKT=PUNKT+1
13175 GOTO 13240
13180 IF ASCII<>45 THEN 13310
13190 IF L>1 THEN 13310
13200 MINUS=MINUS+1
13210 IF ASCII>57 THEN 13220
13220 IF ASCII>47 THEN 13230 ELS
E 13240
13230 ZIFFER=ZIFFER+1
13240 NEXT L
13250 IF PUNKT>1 THEN 13290
13260 IF MINUS>1 THEN 13290
13270 ANZAHL=MINUS+PUNKT+ZIFFER
13280 IF ANZAHL=LAENGE THEN 1331
0
13290 PRINT "EINGABEFEHLER!"
13300 FEHLER=1
13310 RETURN
14000 REM
14010 REM
14020 REM   UP SORTIEREN
14030 REM
14040 REM
14060 FOR I=1 TO SZ
14090 SORT=SZ-I
14100 FOR L=1 TO SORT
14110 IF SF$(I,1)<=SF$(I+L,1)THE
N 14130
14120 GOSUB 14200
14130 NEXT L
14140 NEXT I
14180 RETURN
14200 REM     TAUSCH
14205 FOR K=1 TO 10
14210 H$=SF$(I,K)
14220 SF$(I,K)=SF$(I+L,K)
14230 SF$(I+L,K)=H$
14235 NEXT K
14240 RETURN
15000 REM
15001 REM
15002 REM   UP FARBE/RAHMEN
15003 REM
15004 REM
15005 CALL SCREEN(BF)
15010 PRINT ::::::::::::::::::::
::::
15020 FOR ZS=1 TO 8
15030 CALL COLOR(ZS,2,HF)
15040 NEXT ZS
15050 RETURN
```

Bild 1. Listing des Programms zur Verwaltung einer Bibliothek (Fortsetzung)

zwei beliebige Stichworte eingegeben werden (die zwei ganze Feldinhalte ansprechen). Werden die beiden Stichworte nicht in einem Satz gefunden, erfolgt eine entsprechende Meldung und der Rücksprung ins Menü. Werden die Stichworte innerhalb eines Datensatzes gefunden, wird der Datensatz angezeigt, und es erfolgt eine Abfrage, ob das gefundene Buch auch wirklich das gesuchte Buch ist. Handelt es sich nicht um das gesuchte Buch, wird ab dem nächsten Satz weitergesucht. Ist es das gesuchte Buch, werden am Bildschirm die ersten sechs Feldinhalte zusammen mit einer Wahlliste ausgegeben (auf diese Art und Weise kann man sich also Bücher nach bestimmten Gesichtspunkten heraussuchen lassen, vorausgesetzt natürlich, man hat die Stichpunkte vorher eingegeben). Die Wahlliste ermöglicht ein »Vorwärtsblättern« (ab Zeile 3290) beziehungsweise »Rückwärtsblättern« (ab Zeile 3300) in den Datensätzen, den Ausdruck des Buchtitels, der gerade auf dem Bildschirm zu sehen ist auf einem Drucker (ab Zeile 6000), Änderungen von Feldinhalten in dem Satz des gerade angezeigten Buches (ab Zeile 7000) und einen Rücksprung in die Hauptauswahlliste (Menü).

☐ Katalog (ab Zeile 4000)

Dieses Modul gibt alle Autoren alphabetisch geordnet mit den entsprechenden Buchnummern aus.

☐ Durch Eingabe von »4« kann in die Hauptwahlliste zurückgesprungen werden.

☐ Speichern (ab Zeile 8000)

Mit diesem Modul können die eingegebenen Bücher auf Kassette abgespeichert werden.

Alle Module sind durch eine bestimmte Farbe des Bildschirms mit andersfarbigem Rahmen gekennzeichnet (Unterprogramm ab Zeile 15000).

Die Programmteile Eingabe, Eingabeüberprüfung, Sortieren, Suchen und Farbe sind so gehalten, daß sie auch leicht in anderen (eigenen) Programmen Verwendung finden können.

In Tabelle 1 werden die verwendeten Variablen erklärt. (Rüdiger Vasa)

Liste der Variablennamen

SF$	Variable für den Feldinhalt eines Satzes (zweidimensionale Variable; Dimensionierung erfolgt in Zeile 12, 30 Sätze zu 10 Felder. Anzahl der Sätze ergibt sich aus Probelauf, kann erhöht werden)
FA	gibt an, ob ein Feld alphanumerische Daten (FA=1) oder nur numerische Daten mit maximal einem Punkt und einem Minuszeichen enthält (FA=2)
DN$	Name der Datei (hier: »Bibliotheksverwaltung«)
BF	Hintergrundfarbe
ZS	Zeichensatz
Z	Zustand (CALL KEY(O,T,Z))
T	Taste (wie oben)
SZ	Anzahl der Datensätze
SUCH1$,SUCH2$	Variable für die einzugebenden Stichworte im Suchprogramm
SORT,I,K,L,V	Laufindices
PRUEF$	zu prüfendes SF$ (I,K) im Unterprogramm Eingabeüberprüfung

Tabelle 1. Übersicht über die im Programm verwendeten Variablen-Namen

SHARP
ATARI®
brother
QUALITÄT AUS ERSTER HAND.
WATANABE

MZ-80A 48 KB	2200,—	Atari 400 16K	895,—
MZ-80B 32 KB	3069,—	Atari 800 48K	1895,—
CE-158	448,—	Atari 810	1395,—
Seriell/Parallel		Atari 850	598,—
Schnittstelle für PC-1500		Atari Basic	210,—
PC-1251	339,—	Seikosha GP100A	799,—

Alle Preise in DM inklusive Mehrwertsteuer und voller Garantie.
Bei Versand zuzüglich Versandkosten und Nachnahme.

BAVICO
Vertrieb · Computer · Hardware · Software · Zubehör
Ingolstädter Straße 19, 8071 Großmehring, Tel. (08407) 8598

Meßdaten-Interface für Rechner mit IEC-Bus

DATALOG

Meß- u. Steuerinterface PCI 6300
Flexibles Ein-/Ausgabesystem für analoge Meßgrößen zum Anschluß an Rechner mit IEC-(HP-IB)-Bus

- Standard-IEC-(IEEE-488)-Bus Rechnerschnittstelle
- acht Differenzeingänge 12 Bit, 100 mV, 1 V und 10 V Endbereich softwaremäßig umschaltbar
- Wandlungsrate bis 1000 Mess/s
- vier Analogausgänge, 12 Bit +/– 10 Volt
- vier Relais-Ausgänge zum Schalten externer Geräte
- einfache Programmierung in BASIC
- Lieferung ab Lager

DATALOG
Systeme z. Meßwerterfassung GmbH
Postfach 40, 4053 Jüchen 2
Telefon 0 21 64 - 73 13

MACHEN SIE MEHR AUS IHREM VC-20 UND COMMODORE 64

MEHR Zeichen pro Zeile mit unserem 80-Zeichen-Modul MAXI zur Darstellung von allen VC-20 Zeichen im 25 x 80 = 2000 Zeichen-Format

MEHR Anschlußmöglichkeiten mit unserem neuen IEC-BUS, der optional die BASIC 4.0 Floppykommandos enthält und den Anschluß der gesamten „großen" Commodore Peripheriegeräte bis hin zur Festplatte ermöglicht

MEHR Ausbaumöglichkeiten mit der Modulbox und mit unserem universellen SYSTEM 19 auf Europakartenbasis.

MEHR Druckleistung mit dem SEIKO GP-100 VC und den leistungsstarken EPSON-Druckern mit Original DATA BECKER Interface

MEHR Programmierkomfort mit MASTER für VC-20 und 64 sowie den vielen anderen Programmierhilfen und dem neuen PETSPEED 64 Compiler

MEHR Anwendungsnutzen mit unseren neuen Programmen für Lagerhaltung, Fakturierung, Dateiverwaltung und Textverarbeitung

MEHR Lerneffekt mit den neuesten Lerncassetten zu Mathematik, Biologie, Chemie, Physik und vielen anderen Themen

MEHR Spaß mit den neuesten Spielen aus aller Welt, z.B. Mission Impossible, GRID RUNNER, Adventureland und vielen anderen mehr

MEHR Wissen über Ihren Computer mit unserem riesigen Literaturangebot vom VC-20 ROM Listing über COMPUTE!s First Book of VIC, dem VC-20 Programmer's Reference Guide bis hin zum brandneuen CBM 64 Users Guide

Über unser umfangreiches Angebot erfahren Sie **MEHR,** wenn Sie gegen DM 3,- in Briefmarken unser über 60(!) seitiges VC-Info 1/83 anfordern oder uns demnächst einmal in Düsseldorf besuchen. In unserem 800 qm großen Ausstellungszentrum finden Sie nicht nur eine riesige Auswahl rund um den VC-20 und CBM 64, sondern auch so bekannte Namen wie ATARI, BASIS, EPSON, COMMODORE, HEWLETT PACKARD, GENIE, OSBORNE, SHARP und SIRIUS. DATA BECKER, eine Tochter von AUTO BECKER und im selben Gebäude wie das weltbekannte Autohaus, ist die weiteste Reise wert.

IHR GROSSER PARTNER FÜR KLEINE COMPUTER

DATA BECKER

Merowingerstraße 30 · 4000 Düsseldorf 1 · Telefon 0211/312085 · Telex 08582874

```
0 REM ************************************************************
1 REM *  CMD-BASIC-MODUL 2 :      BERECHNETES 'RESTORE'           *
2 REM *  ------------------------------------------------------  *
3 REM *  9.AUG 1982 - ULF BARTELT      ! ALTE CMD-FUNKTIONEN     *
4 REM *              JAHNSTR. 77A      ! WIE  CMD"T" BLEIBEN     *
5 REM *        D-4133 NEUKIRCHEN-VLUYN ! ERHALTEN.               *
6 REM ************************************************************
7  CLS
  :MS=PEEK(16561)+256*PEEK(16562)
  :PRINT"MEMORY SIZE, ALTER WERT :";MS
  :MS=MS-30
  :PRINT"MEMORY SIZE, NEUER WERT :";MS
  :POKE 16562,FIX(MS/256)
  :POKE 16561,MS-256*FIX(MS/256)
  :CLEAR 50
  :MS=PEEK(16561)+PEEK(16562)*256
8  FOR AD=MS+1 TO MS+30
  :IF AD>32767
      THEN A=AD-65536
      ELSE A=AD
9  READ X$
  :IF LEFT$(X$,1)="("
      THEN X=PEEK(VAL(MID$(X$,2,5)))
      ELSE X=VAL(X$)
10 POKE A,X
  :NEXT
  :MS=MS+1
  :POKE 16755,195
  :POKE 16756,MS-256*FIX(MS/256)
  :POKE 16757,FIX(MS/256)
  :PRINT
  :PRINT"'CMD RESTORE <INTEGERAUSDRUCK>' IST INITIALISIERT"
  :DELETE 0-11
11 DATA
    229, 43,215,254,144, 40,  4,225,(16755),(16756),(16757),241,
     35,205,  2, 43,229,205, 44, 27, 11,237, 67,255, 64,225,216,
    195,217, 30
```

Bild 2. Das Programm poked die RESTORE-n-Routine in den Speicher und löscht sich dann selbst

Noch eine Anmerkung: Durch die Verwendung der ROM-Routine zur Analyse eines Integerausdruckes ist es nötig, Zeilennummern über 32767 umzurechnen. Solche Werte müssen um 65536 erniedrigt werden. Zum Beispiel:
IF A>32767 THEN A=A-65536 oder
A=A+65536*(A>32767).

Die zweite Form hat den Vorteil, daß die Zeile um weitere Anweisungen erweitert werden kann, die in jedem Fall ausgeführt werden — ob die Adresse nun umgerechnet wird oder nicht.

Zum Verständnis dieser Anweisung: Der logische Vergleich in der Klammer kann wahr oder falsch sein. In Basic wird »wahr« durch den Wert —1 (FFFF Hex auf Maschinenebene), »falsch« dagegen durch 0 ausgedrückt. Es wird also entweder —65536 zur Variablen A hinzugezählt oder Null. Aber es dürfte wohl keinem Basic-Programmierer schwerfallen, mit den Zeilennummern unter der Grenze von 32767 zu bleiben.

(Ulf Bartelt, Andreas Schwaller)

Benutzerführende Menü-Steuerung

Manchen Anwender stört es, daß in der Menü-Technik für den TI 99/4(A) auch in Extended Basic nur drei Möglichkeiten der Eingabe-Anforderung existieren: Erstens INPUT ohne Eingabeprüfung, zweitens ACCEPT mit VALIDATE-Option (zur Eingabeprüfung) sowie drittens CALL KEY.

Während ACCEPT zwar die Vorgabe eines Prüfstrings erlaubt, benötigt es zwei Tasten zur Eingabe (die gewählte Ziffer oder das Zeichen sowie ⟨ENTER⟩). Eine Eingabeprüfung bei CALL KEY ist nur

TI 99/4(A)

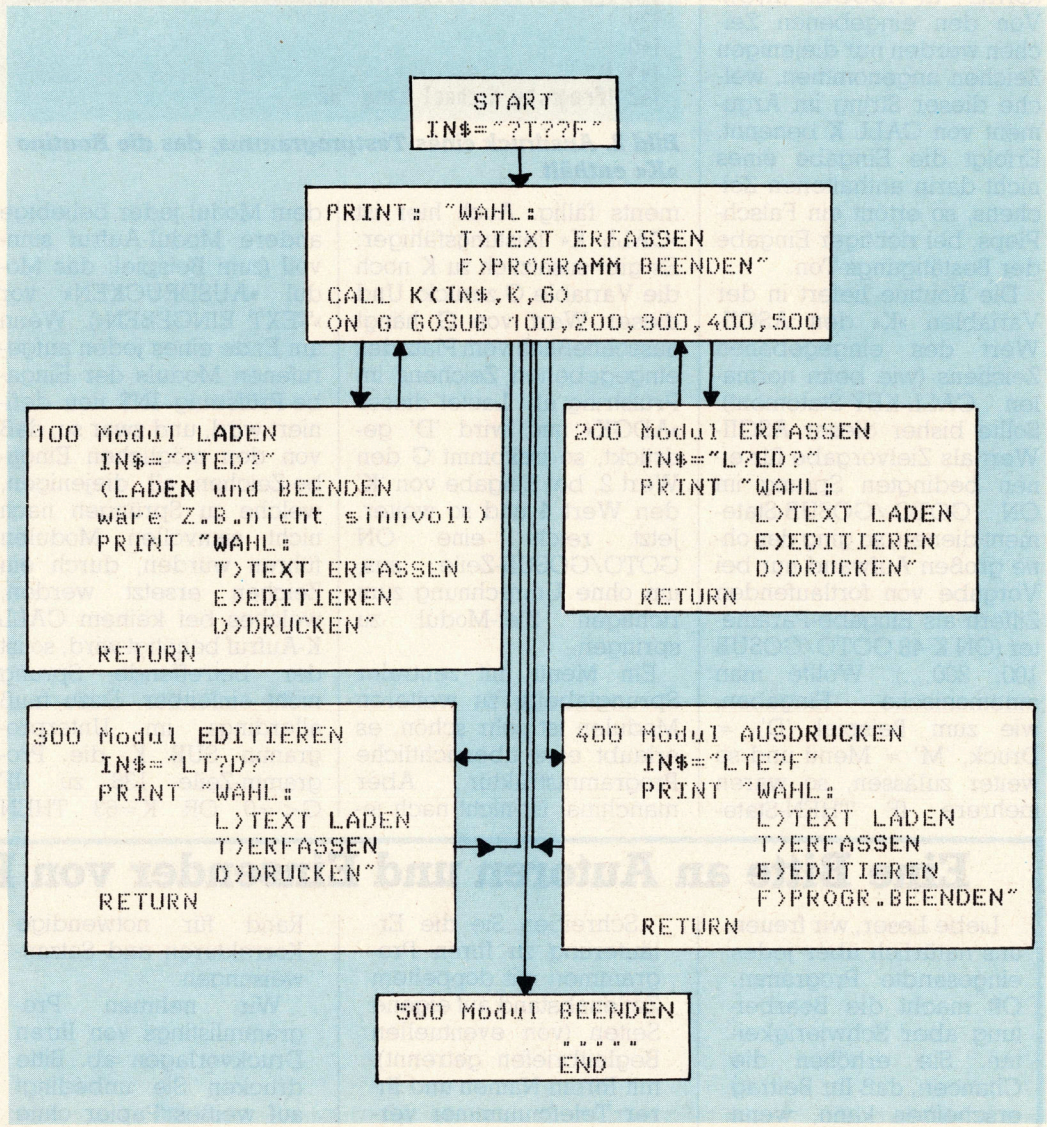


Bild 1. Das Diagramm verdeutlicht die Steuerung der Eingabekontrolle durch Ändern des Strings »IN$«

```
100 REM VARIABLE:
101 REM
102 REM IN$....Alle in diesem String vorhandenen Zeichen
103 REM          und nur diese werden von SUB K akzep-
104 REM          tiert!
105 REM K......Rückgabe: ASCII-Wert des eingetippten
106 REM          Zeichens
107 REM G......Rückgabe: Stellenwert des gedrückten Zei-
108 REM          chens in IN$ (z.B. für ON GOTO oder
109 REM          ON GOSUB
110 !
111 !
112 !
113 REM ANWENDUNGSBEISPIEL:
114 CALL CLEAR
115 IN$="ABCXYZ"
116 CALL K(IN$,K,G):: ON G GOSUB 122,123,124,127,127,127
117 GOTO 116
118 !
119 !
120 !
121 REM ANWENDUNG VON VARIABLE G
122 PRINT "'A' WURDE GEDRUECKT." :: RETURN
123 PRINT "'B' WURDE GEDRUECKT." :: RETURN
124 PRINT "'C' WURDE GEDRUECKT." :: RETURN
125 !
126 REM ANWENDUNG VON VARIABLE K
127 PRINT "*"&CHR$(K)&"* WURDE GEDRUECKT." :: RETURN
128 !
129 !
130 !
131 REM ************************************************
132 SUB K(IN$,K,G)
133 IF F=0 THEN CALL SOUND(150,1400,5)ELSE F=0
134 CALL KEY(3,K,R):: IF R<1 THEN 134
135 G=POS(IN$,CHR$(K),1)
136 IF G<=0 THEN CALL SOUND(200,220,0):: F=1 :: GOTO 133
137 SUBEND
138 REM ************************************************
139 !
140 !
141 !
142 !Progr.by Michael Lang '82
```

Bild 2. Ausdruck eines Testprogramms, das die Routine »K« enthält

durch eine IF THEN-Zeile möglich, ohne die Kontrolle des Richtig-Falsch-Piepses, wie ihn ACCEPT bietet. Dafür bestünde der Komfort einer Ein-Tasten-Eingabe. Um alle Eigenschaften zu vereinen (Eingabeprüfung im CALL-Statement, Richtig-Falsch-Pieps und Ein-Tasten-Eingabe), wurde das Programm für die Sub-Routine »CALL K« geschrieben.

Diese Routine bietet sich vor allem zum Einsatz in menüartigen Programmstrukturen an. Eine Eingabeprüfung findet durch die Vorgabe eines Strings im CALL-Statement statt. Man kann eine Stringvariable im Argument angeben, (zum Beispiel CALL K(A$,K,G), wobei vorher A$="123456" definiert wurde), oder einen absoluten String setzen (CALL K("ABCDEF",K,G)). Von den eingebenen Zeichen werden nur diejenigen Zeichen angenommen, welche dieser String im Argument von CALL K benennt. Erfolgt die Eingabe eines nicht darin enthaltenen Zeichens, so ertönt ein Falsch-Pieps, bei richtiger Eingabe der Bestätigungs-Ton.

Die Routine liefert in der Variablen »K« den ASCII-Wert des eingegebenen Zeichens (wie beim normalen CALL-KEY-Statement). Sollte bisher dieser ASCII-Wert als Zielvorgabe für einen bedingten Sprung im ON GOTO-/GOSUB-Statement dienen, so ging das ohne großen Aufwand nur bei Vorgabe von fortlaufenden Ziffern als Eingabe-Parameter (ON K-48 GOTO/GOSUB 100, 200,...). Wollte man mnemonische Eingaben, wie zum Beispiel 'D' = Druck, 'M' = Menü und so weiter zulassen, so waren mehrere IF THEN-Statements fällig. Auch hier ist »CALL K« leistungsfähiger: Es gibt zusätzlich zu K noch die Variable G zurück. Und dieser Wert von G hängt ausschließlich vom Platz des eingegebenen Zeichens im Prüfstring ab. Lautet dieser »ADGK« und wird 'D' gedrückt, so bekommt G den Wert 2, bei Eingabe von 'K' den Wert 4 und so weiter. Jetzt reicht eine ON GOTO/GOSUB-Zeile aus, um ohne Umrechnung zum richtigen Ziel-Modul zu springen.

Ein Menü mit zentraler Sprungtabelle zu weiteren Modulen ist sehr schön, es erlaubt eine übersichtliche Programmstruktur. Aber manchmal ist nicht nach jedem Modul jeder beliebige andere Modul-Aufruf sinnvoll (zum Beispiel: das Modul »AUSDRUCKEN« vor »TEXT EINGEBEN«). Wenn am Ende eines jeden aufgerufenen Moduls der Eingabe-Prüfstring IN$ neu definiert wird, und zwar so, daß von den möglichen Eingabe-Zeichen all diejenigen, welche zu Sprüngen nach nicht sinnvollen Modulen führen würden, durch ein Zeichen ersetzt werden, welches bei keinem CALL K-Aufruf benötigt wird, so ist der betreffende Sprung nicht einleitbar. Dazu muß allerdings im Unterprogramm SUB K die Programm-Zeile 136 zu »IF G<=0 OR K=63 THEN CALL SOUND (200,220,0) ::F=1:: GOTO 133« geändert werden. So läßt sich trotz einer einzigen Sprungtabelle an zentraler Stelle eine sinnvolle Ablaufsteuerung erreichen. Bild 1 zeigt ein Beispiel aus der Textverarbeitung. Durch die entsprechende IN$-Definition mit dem »tabuisierten« Fragezeichen (oder jedem in Zeile 136 definierten anderen Zeichen) an geeigneter Stelle, wird ein solcher unlogischer Sprung verhindert, obgleich nur eine einzige ON GOSUB-Sprungtabelle für alle Aufrufe existiert.

Für die Sprungsequenz ist einzig und allein die Beziehung der Zeichenfolge in IN$ zur Sprungtabelle in ON GOSUB entscheidend. Erfolgt die Bildschirm-Ausgabe des Menüs jeweils am Schluß des eben abgearbeiteten Moduls, so kann die Menüliste auf die Eingaben abgestimmt werden, die nach Rückkehr zum zentralen ON GOSUB-Statement tatsächlich möglich sind.

In Programm-Zeile 134 erfolgt ein CALL KEY(3,K,R)-Statement. Die Ziffer »3« im Argument bewirkt eine Umschaltung der Tastatur in den Großbuchstaben-Modus, der auch nach Verlassen der Sub-Routine erhalten bleibt. Will man, daß nach CALL K noch Kleinbuchstaben angenommen werden, so muß die Zeile 137 wie folgt abgeändert werden:
137 CALL K(5,R,R):: SUBEND.

Sollen schon im CALL K-Aufruf Kleinbuchstaben angenommen werden, so ist Zeile 134 so zu formulieren:
134 CALL KEY(5,K,R)::.....

(Michael Lang)

Eine Bitte an Autoren und Einsender von Programmen:

Liebe Leser, wir freuen uns natürlich über jedes eingesandte Programm. Oft macht die Bearbeitung aber Schwierigkeiten. Sie erhöhen die Chancen, daß Ihr Beitrag erscheinen kann, wenn Sie folgende Punkte beachten:

Schreiben Sie die Erläuterung zu Ihren Programmen mit doppeltem Zeilenabstand auf eigene Seiten (von eventuellen Begleitbriefen getrennt), mit Ihrem Namen und Ihrer Telefonnummer versehen. Lassen Sie links und rechts je drei cm Rand für notwendige Korrekturen und Satzanweisungen.

Wir nehmen Programmlistings von Ihren Druckvorlagen ab. Bitte drucken Sie unbedingt auf weißes Papier ohne Linienvordruck. Verwenden Sie notfalls die Rückseite Ihres Druckpapiers, wenn Sie kein Papier ohne Linierung haben — und arbeiten Sie bitte mit einem neuwertigen Farbband.

Vielen Dank, besonders für die dadurch ersparten Mühen und Rückfragen!

Grafische Bildschirm-Hardcopy

TI 99/4(A)

Anders als bei dem in Heft 1/83 veröffentlichten Programm von Walter Eckert, das nur Standardzeichen zu kopieren in der Lage ist, kann das folgende Programm jeden beliebigen Bildschirminhalt kopieren.

Da mit dem TI99/4A ohne große Schwierigkeiten umfangreiche Grafikdarstellungen möglich sind, erwächst bald der Wunsch, eine Hardcopy des Bildschirms zu erhalten. Um dies mit dem Drucker GP 100A möglich zu machen, ist eine umfangreiche Umcodierung der auf dem Bildschirm in einer 8x8-Punktmatrix vorliegenden Grafik auf das 5-7-Matrixformat des GP 100A zu vollziehen. Das vorliegende Programm (Bild 1) kann als Unterprogramm eingesetzt werden und ist in TI-EXTENDED BASIC geschrieben. 7 Bildschirmzeilen werden dabei in 8 Druckerzeilen umgewandelt. Die Durchführungsdauer hängt von der Grafikdarstellung ab, das angegebene Beispiel benötigt knapp 40 Minuten. Als Nebenprodukt des Programms erhält man eine weitere Schriftsorte am Drucker GP 100A, da die beim TI99/4A am Bildschirm als kleinere Großbuchstaben wiedergegebenen Kleinbuchstaben in gleicher Weise zu Papier gebracht werden. Bei der Erstellung der Grafik muß dabei darauf geachtet werden, keine ASCII-Zeichen größer als 140 zu verwenden, da diese im EXTENDED BASIC für Sprites reserviert sind. Bild 2 zeigt eine typische Wiedergabe mit Grafik. (Karl Hagenbuchner)

Bild 1. Ausdruck des Hardcopy-Programms

```
10000 DIM N$(32)
10010 FOR L=0 TO 3
10020 FOR SP=1 TO 32
10030 FOR Z=7*L+1 TO 7*L+7 :: IF Z>24 THEN 10060
10040 CALL GCHAR(Z,SP,T):: IF T<>31 THEN CALL CHARPAT(T,H$)ELSE H$=RPT$("0",16)
10050 N$(SP)=N$(SP)&H$
10060 NEXT Z :: IF LEN(N$(SP))<112 THEN N$(SP)=N$(SP)&RPT$("0",112-LEN(N$(SP)))
10070 NEXT SP
10080 OPEN #255:"PIO.CR" :: PRINT #255:CHR$(08);
10090 FOR Z=1 TO 8 :: PRINT #255:CHR$(27);CHR$(16);CHR$(00);CHR$(111);
10100 FOR SP=1 TO 32 :: FOR H=1 TO 8 :: Y(H)=128 :: NEXT H :: IF SEG$(N$(SP),14*
(Z-1)+1,14)=RPT$("0",14)THEN 10160
10110 FOR H=1 TO 7 :: IF L=3 AND Z=5 THEN 10180
10120 H$=SEG$(N$(SP),14*(Z-1)+2*H-1,1):: G$=SEG$(N$(SP),14*(Z-1)+2*H,1)
10130 IF ASC(H$)>60 THEN H$=STR$(ASC(H$)-55)
10140 IF ASC(G$)>60 THEN G$=STR$(ASC(G$)-55)
10150 B=VAL(H$):: I=1 :: GOSUB 10200 :: B=VAL(G$):: GOSUB 10200 :: NEXT H
10160 PRINT #255:CHR$(Y(1));CHR$(Y(2));CHR$(Y(3));CHR$(Y(4));CHR$(Y(5));CHR$(Y(6
));CHR$(Y(7));CHR$(Y(8));
10170 NEXT SP :: PRINT #255:CHR$(10):: NEXT Z
10180 FOR SP=1 TO 32 :: N$(SP)="" :: NEXT SP :: CLOSE #255 :: NEXT L
10190 RETURN
10200 FOR J=3 TO 0 STEP -1
10210 IF B>=2^J THEN Y(I)=Y(I)+2^(H-1):: B=B-2^J
10220 I=I+1 :: NEXT J
10230 RETURN
```

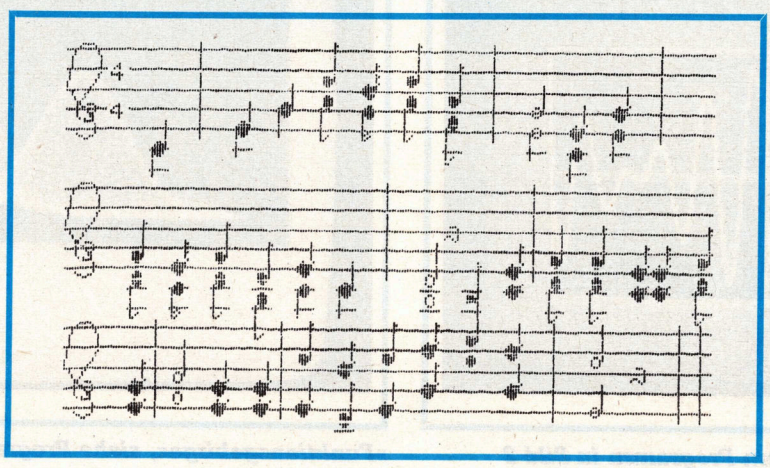


Bild 2. Gedruckte Wiedergabe einer Bildschirmgrafik

DR. AUMANN GmbH
Computersysteme

VIDEO GENIE 3003 mit BASIC3,
16k RAM, Groß-/Kleinschrift komplett 1450 DM
mit 9 Zoll Monitor 1650 DM
EG 3008 (GENIE II) 1550 DM
48k Speichererw. 180 DM
Floppycontrollerkarte, Druckeranschluß, Uhr ohne RAM, unbestückte Platine/TRS80 100 DM
komplett bestückt **450 DM**
komplett mit Doubler 650 DM
EG 3014 32k Exp.Interface mit Floppy-, Druckerinterface 1250 DM
Groß-/Kleinschriftmodul, deutsche Zeich. f. 3003 140 DM
EG3003 Platinensatz ohne Tastatur/Netzteil 650 DM
COLOR Genie 16k 950 DM
GENIE III 64k, 2 Floppys 6500 DM
16k Speichererweiterung 200 ns TRS-80 u. Exp.Interface 40 DM

Floppylaufwerke für TRS-80 VIDEO GENIE, Sorcerer, APPLE etc.
BASF 6106 40 Sp. SS/SD 490 DM
BASF 6106 40 Sp. SS/DD 575 DM
Siemens 40 Sp. SS/DD 700 DM
MPI/Shg 2x40 Sp. DS/DD 850 DM
SHUGART 80 Sp. SS/DD 1190 DM
YE DATA 2x80 Sp. DS/DD 1351 DM
Doppelgeh./2x Netzteil 230 DM
YE DATA 2x77 Sp. 8" DD 1790 DM
Doppelgeh. NT 2x8" 490 DM

Hard Disk für APPLE, MINIFRAME XEROX 820, SASI Bus 10 MioByte mit Controller kompl. 7990 DM

Der Speicher-Star komplett anschlußfertig
Doppelfloppy für TRS-80/GENIE 2x40 Tr.Doppelkopf, NEWDOS80/2, deutsches Manual, Doubler, Kabel
über 730 KByte nur 2150 DM
oder 1400 KByte nur 3150 DM

DISKETTEN
Preise gelten je Stück wobei 1. Preis 10 Stk, 2. Preis 100 Stk
DYSAN(DYPY) 5" 6,50 / 5,60 DM
DYSAN(DYPY) 8" 6,40 / 5,63 DM
3 Musterdisk gegen 20 DM-Schein, VERBAT.DATALIFE mit Innenring und 5 Jahren Garantie
VERBATIM 5" DD 7,70 / 6,92 DM
VERBATIM 8" SD 7,90 / 7,00 DM
BASF SS/SD 5" 7,00 / 6,31 DM

Monitor 12 Zoll, grün, 18 MHz, sehr gute Qualität 550 DM
Monitor 12 Zoll gr. 290 DM
C.ITOH 8510A Matrixdr. 1850 DM
dito 1550 Matrixdrucker für breites Papier 2250 DM
STAR Matrixdrucker 80 Z/Zeile, 80 Z/s Groß/Klein 950 DM
Typenraddrucker ITOH 45 Z/s breiter Wagen 4500 DM

——ACHTUNG——
>>>Einplatinenrechner<<<
OEMs, Systemhäuser, kommerz. Benutzer und Spezialisten: MINIFRAME1 Z80A, 64k Computer, 2x RS232, Video 80x24, Druckerinterface Centronics, 8k ROM, Monitor im ROM, CP/M 2.2 Betr.system mit Bildschirm/deutsch. Tastatur, 8" Floppylaufwerk, Komplettpreis o. CP/M 6989 DM
Made in Germany
Leiterplatte dito. unbestückt mit Manuals 500 DM
In Kürze MINIFRAME1 2x 5" Disk Komplettgerät 6500 DM
CP/M 2.2 incl. CBIOS 550 DM
TINYBASIC im ROM mit Diskbefehlen (SAVE/LOAD etc) 200 DM
FORTH im ROM (FIG FORTH) mit Diskbefehlen, Editor, Compiler Decompiler etc. 370 DM
C-Compiler f. CP/M mit Bildschirmeditor (Small C) 230 DM

MINIFRAME4 Z80A, 128k, Floppycontroller f. 4x 5" und 4x 8", DMA zu und von RAM <—> Periph. 2x RS232, 1x parallel, Harddisk Interface, bestückte und geprüfte Leiterplatte 3158 DM

Compulady IV 64k 3000 DM

Z8 BASIC Controllcomp. 895 DM
IBM PC kompatibel 16bit 8088
256k, 9 Slots, Arith.Processor, Floppycontroller 5", Centr. u. V.24, Leerplatine 1250 DM
komplett bestückt 7900 DM

68000 16/32 Bit Rechnerplatine
unbestückt, RAM, EPROM, Interfaces parallel, V.24 420 DM
256k RAM, 64k EPROM, Floppyinterface in Vorbereitung 2/83, OEM- und Händleranfragen willkommen. Fordern Sie unsere Liste an über Software und gebrauchte Geräte.
Wir sind Vertriebspartner für **RANK-XEROX** Computersystem 820
Alle Preise incl. MwSt. ab Werk. 6 Monate Garantie

DR. AUMANN GmbH
Computersysteme
Schulstr. 12, 5451 Strassenhaus
Tel. 02634-4592

Der Heimcomputer-Markt explodiert

300000 Heimcomputer in Europa, 500000 in den USA — das sind die Zahlen von 1981. Chuck Digate, Geschäftsführer des Home-Computer-Bereiches bei TI in Europa, erwartet einen wahnwitzigen »Run« auf diese »Unter-500-Dollar-Geräte«: 1983 sollen 2,4 Millionen davon in Europa und 5,7 Millionen auf dem US-Markt abgesetzt werden.

Daß Texas Instruments mit dem nun um 20 Prozent verbilligten TI 99/4A und seinen Neuvorstellungen TI 99/2 und CC40 an diesem Boom teilhaben will, ist wohl selbstverständlich — und daß sich in Europa 1983 nicht 2,4 Millionen Freaks, Hobbyisten und Bastler als mögliches Kundenpotential für Heimcomputer rekrutieren lassen, ist wohl auch klar. Also muß sich die Käuferstruktur gewandelt haben. Nach Untersuchungen von Texas Instruments interessieren sich »ganz normale Familien« dafür, was eigentlich ein Computer ist. Studien hätten gezeigt, daß bei 50 Prozent dieses Interesse sich in dem Wunsch äußert, zu verstehen, wie so ein Computer funktioniert. Der Informatik-Unterricht in der Schule tue ein übriges: Auch Eltern möchten Computersprachen ebenso verstehen wie ihre Kinder.

So gilt denn nach H.J. Poertner, European Product Marketing Manager Homecomputer bei Texas Instruments: »Der typische Entscheider zur Zeit für den Kauf eines Homecomputers in der Familie ist männlich, 31 bis 40 Jahre alt, Ausbildung etwas über dem Durchschnitt, verheiratet, 3 bis 4 Familienmitglieder. Das jährliche Haushalts-Nettoeinkommen ist leicht überdurchschnittlich. 86 Prozent der Käufer sind männlich, wenngleich die weiblichen Mitglieder der Familie an der Kaufentscheidung teilhaben. 75 Prozent sind verheiratet. Wenn man die Altersgruppen generell betrachtet, sehen wir, daß 80 Prozent aller Käuferzwischen 19 und 50 Jahren sind. Der Anteil der 31-bis 40jährigen liegt etwa bei 38 Prozent. Diese Daten sind aus dem November 82. Wenn man diese Zahlen mit einer Untersuchung von Anfang März dieses Jahres in Deutschland vergleicht, zeigt sich eine Verschiebung: 17 Prozent der Käufer waren — nach dieser Stichprobe im März — unter 16 Jahre alt, 19 Prozent zwischen 16 und 24 Jahren und 38 Prozent zwischen 25 und 40 Jahren.«

Der europäische Heimcomputermarkt entwickelt sich von zwei Seiten: der ZX81-Seite, also billige einfache Geräte, die den Anwender sehr bald an die Grenze ihrer Leistungsfähigkeit stoßen lassen, und der Video-Spiel-Seite, wobei die Video-Spiele nicht für ernsthafte Home-Anwendungen, Lernen und Programmieren eingesetzt werden können. Der ideale Homecomputer soll Spielen und Lernen, Homeanwendungen sowie Programmieren erlauben —, das zeigt auch die Tabelle 1, die die Gründe der Kaufentscheidung für einen Homecomputer widerspiegelt. Daß — intelligente — Spiele nach einem halben Jahr den Hauptanteil des Computereinsatzes ausmachten, mag angesichts der »Fernsehmüdigkeit« nicht verwundern. Das US-Angebot an Spielen für Apple II und den IBM-Personal Computer beweist, daß sogar sogenannte Business-Computer gerne auch zum Spielen eingesetzt werden. So ist es dann auch nicht verwunderlich, wenn die Homecomputer-Hersteller auf Spiele setzen. Andererseits werden sich Homecomputer in Zukunft nur dann gut verkaufen, wenn Lernprogramme und Software für »ernsthafte Anwendungen« in ausreichendem Maße — möglichst in deutsch — zur Verfügung stehen. Auf diesem Sektor tun zumindest die Großen der Homecomputer-Branche — Atari, Commodore, Sinclair und Texas Instruments — einiges: allein Atari will in diesem Jahr 3 Millionen Mark in die Entwicklung deutscher Software stecken.

Der ideale Heimcomputer soll zum Spielen, Lernen und Programmieren eingesetzt werden können

Texas Instruments glaubt mit dem TI 99/4 und dem neuen TI 99/2 insgesamt den Umsatz in Deutschland verachtfachen zu können. Dabei ist der TI 99/2 eine »Ergänzung des TI 99/4 nach unten«, wie TI-Marketingdirektor W. Glöckle, zuständig für Deutschland, hervorhob. Der als direkte Konkurrenz zum Sinclair ZX81 konzipierte TI 99/2 soll unter 300 Mark kosten und ab dem 3. Quartal dieses Jahres lieferbar sein. Bestückt ist er mit dem 16-Bit-Prozessor TI 9995, 24 KByte ROM mit Basic-Interpreter und Betriebssystem sowie 4,2 KByte RAM (erweiterbar auf 36,2 KByte). Angeschlossen werden kann jedes Fernsehgerät, auf dem 24 Zeilen zu je 28 Zeichen darstellbar sind. Im Gegensatz zum ZX81 wird keine Folientastatur, sondern eine »richtige« QWERTY-Tastatur verwendet, deren Tasten — wie beim Spectrum — aus Hartgummi bestehen. Der TI 99/2 arbeitet wie der TI 99/4 mit TI-Basic, so daß also bei Basic-Programmen Kompatibilität zwischen diesen beiden Texas-Home-Computern besteht. An Software werden neben Programmen auf Kassette auch Programm-Module angeboten, die einfach in den TI 99/2 eingesteckt werden. Zwei davon mit den Titeln »Learn to program« und »Learn to

Der Compact Computer CC40 mit den neuen Peripheriegeräten (rechts): 4-Farb-Printer/Plotter, Wafertype-Drive und RS232-Interface

program Basic« sind speziell für den Computer-Neuling gedacht — Englisch-Kenntnisse werden allerdings vorausgesetzt.

Neben Kassettenrecordern (Texas Instruments bietet einen neuen 198-Mark-Recorder an, zu dem ein Anschlußkabel für den TI 99/4 mitgeliefert wird; das Anschlußkabel für den 99/2 liegt diesem bereits bei) können auch Peripheriegeräte der TI-Hex-Bus-Familie angeschlossen werden. Dies sind zur Zeit:

☐ Ein Wafertape-Drive, häufig auch als Stringy-Floppy bezeichnet, mit einer Speicherkapazität von 48 KByte. Als Speichermedium werden Endlos-Bandkassetten eingesetzt. Während »konventionelle« Kassettenrecorder zum Laden eines 32 KByte großen Programmes 2 bis 5 Minuten brauchen, sollen es Stringy Floppys in zirka 12 Sekunden schaffen. Der Preis für den Wafertape-Drive wird bei zirka 600 Mark liegen;

☐ ein 4-Farben-Printer/Plotter, der es erlaubt, Hardcopys auch von Grafiken zu erstellen. Maximal 36 Zeichen können in eine Zeile gedruckt werden. Verwendet wird 2½ Zoll breites Normalpapier. Voraussichtlicher Preis: ungefähr 700 Mark;

☐ ein RS232-Interface, das um die 200 Mark kosten soll, erlaubt den Anschluß von Standard-Druckern, anderen Peripheriegeräten oder gar anderen Computern.

Diese drei Peripherieeinheiten sind ebenfalls an die zweite Texas-Instruments-Neuheit, den tragbaren CC40 anschließbar, der 748 Mark kostet und ab April dieses Jahres lieferbar sein soll. Die wichtigsten Merkmale sind: LCD-Anzeige (1 Zeile mit 31 Zeichen), QWERTY-Tastatur mit separatem Zahlenfeld, 8-Bit-CMOS-Prozessor 70C20, 6 KByte RAM, Betriebssystem mit Basic-Interpreter in 34 KByte ROM sowie ein Anschluß für weitere 12 KByte RAM oder Software-Module mit bis zu 128 KByte ROM. Die Abmessungen des CC40 betragen 240 mm x 145 mm x 25 mm, das Gewicht 600 g. Eingesetzt werden Alkali-Mangan-Batterien, die eine Betriebsdauer von zirka 200 Stunden gewährleisten sollen. Der CC40, für den Fernsehinterface und Barcodeleser bereits in Entwicklung sind, ist laut K. Nielsen, zuständig für den Vertrieb des CC40, zwischen dem Sharp PC 1500 und dem Epson HX20 anzusiedeln. Einsetzen soll ihn der Geschäftsmann auf Reisen; aber auch zur mobilen Datenerfassung und in Schulen sei er prädestiniert: Der Schüler kann Gerät und Programm mit nach Hause nehmen und weiter an dem im Unterricht begonnenen Programm »herumbasteln«.

Unter 300 Mark: Der TI 99/2 — eine direkte Konkurrenz zum ZX81?

20 Prozent Preissenkung »auf Grund der rasch gestiegenen Absatzzahlen« gelten seit dem 1. April 1983 für den TI 99/4. Der Preis wird damit unter 650 Mark liegen.

Neue Computer und Preissenkungen sind eine Voraussetzung für ein erfolgreiches Agieren auf dem Homecomputer-Markt — doch verkauft werden müssen die Geräte trotz alledem. Dazu will Texas Instruments auch seine Vertriebsstrategie ändern. Während früher die Fachhändler im Vordergrund standen, will man zukünftig verstärkt über den »Braune-Ware«-Markt, also Radio- und Fernsehhandel sowie Warenhäuser gehen. Die Struktur der Absatzwege zeigt Tabelle 2.

Aber warum soll man überhaupt einen Heimcomputer kaufen? Bei Texas Instruments argumentiert man so:

Der Homecomputer soll sowohl von Kinderbeinen an Hilfsmittel für den beruflichen Werdegang sein als auch eine Alternative zum Fernsehgerät und Videorecorder darstellen. Im Zusammenhang mit Bildschirmtext wird der Homecomputer als intelligentes Ein-Ausgabegerät fungieren.

Um die Homecomputer den potentiellen Käufern näherzubringen, will man nicht nur den Bekanntheitsgrad durch Fernsehwerbung steigern, sondern auch Beratung und Unterstützung des Kunden verbessern. Dazu gehört neben speziell von Texas Instruments ausgebildetem Personal, das sowohl Käufer als auch Verkäufer in den einzelnen Verkaufsstellen unterstützten soll, vielfältige und anwenderfreundliche Software sowie Kurse und Literatur zum Erlernen von Basic: Neben einem Fernkurs der Studiengemeinschaft Darmstadt können diejenigen, die TI-Basic und TI-Extended-Basic nicht aus einem in Kürze erscheinenden Buch lernen wollen, dies ab Herbst '83 in kostenlosen 3-Stunden-Kursen tun. Das Unterrichts-Material soll so gut sein, daß Erwachsene in dieser Zeit kleinere Basic-Programme selbst schreiben könnten.

Neben 6 Monaten Garantie soll dem Kunden ein besonderer Service geboten werden: Der Händler hat die Möglichkeit, das defekte Gerät des Kunden sofort gegen ein funktionierendes einzutauschen, so daß der Kunde keine Reparaturzeiten in Kauf nehmen muß. Der Händler erhält dann direkt von Texas Instruments ein neues Gerät im Austausch gegen das alte. (sc)

Der Computer wurde erworben zum	Angabe zum Zeitpunkt des Kaufs	Angabe ein halbes Jahr nach dem Kauf
Spielen	24 %	45 %
Lernen (allgemeine Fächer wie Rechtschreibung, Fremdsprachen, Mathematik usw.)	30 %	23 %
Heimanwendungen (wie Haushaltsbuchführung, Textverarbeitung etc.)	20 %	10 %
Programmieren (lernen)	26 %	22 %

Tabelle 1. Mehr gespielt als geplant: Das ist das Ergebnis einer Umfrage, die Texas Instruments bei Familien durchführen ließ, die einen TI 99/4 besitzen. Ein halbes Jahr nach dem Kauf zeigte sich, daß der Heimcomputer viel häufiger zum Spielen verwendet wurde, als man ursprünglich wollte.

Warenhäuser	30 %
Fachhandel (Büro-Schreibmaschinenhandel)	20-25 %
Radio- und Fernsehhandel	30 %
Computershops	10-15 %

Tabelle 2. Über diese Kanäle will Texas Instruments seine Homecomputer absetzen

Trinkt nur noch ab und zu ein Weißbier: Hobby-Programmierer Rüdiger Neumann in einer Münchner Gaststätte

Gesund bleiben: weniger rauchen

Wer über einen längeren Zeitraum hinweg Zahlenwerte, zum Beispiel für Alkohol- oder Nikotinkonsum, Kalorien oder sportliche Leistungen erfassen, auswerten und ihre Veränderung analysieren will, ist gut mit einem grafischen Programm, vielleicht mit Balken, bedient. Nicht nur der Mensch, der sich Dinge, die das Leben erst schön machen, abgewöhnen will (oder muß), findet eine für ihn wertvolle moralische Krücke in der ständigen Beobachtung der täglichen Veränderung zum Besseren hin.

Da es mir mit einem »Säuferbalkenprogramm« gelungen ist, das regelmäßige Trinken durch ebenso regelmäßige Computer-»Überwachung« abzustellen (siehe CP 24/82), nahm ich es für meinen Freund Kurt zu Hilfe, dem sein täglicher Zigarettenkonsum lästig und teuer geworden war (seiner nichtrauchenden Freundin nicht weniger!).

Jeden Montag brachte er mir brav seine Strichliste, und ich brachte ihm am nächsten Tag sein gedrucktes Sünden-Diagramm. Tag für Tag fügten sich die Einzelwerte zu einem untrüglichen Spiegelbild seiner Gewohnheit. Zum Jahresende beschloß er daraufhin, das Rauchen schlagartig ganz zu lassen. »Jetzt brauch' ich deinen blöden Computer nicht mehr!«

Das Diagramm zeigte, daß es ihm aus eigener Kraft gelungen war, auf die Hälfte seiner bisherigen Werte über einen Zeitraum von mehreren Monaten zu kommen. Es ist vermessen, zu behaupten, das habe der Computer vollbracht. Aber er hat durch das tägliche »Vor-Augen-Führen« sicher einen wichtigen Teil dazu beigetragen.

Jetzt zur technischen Seite. Der TI 99/4 (mit Diskettenstation, Modul Extended Basic und Speichererweiterung) hat 24 Zeilen à 32 Zeichen. Es liegt also auf der Hand, in den 32 Spalten genau einen Monat samt Maßstab unterzubringen. Wir benutzen von den 24 Zeilen 13 für die Säulen, die restlichen für Überschriften, Unterstreichungen, Darstellungen von Sonderzeichen (Wochentage) und dergleichen. Die unterste Zeile ist für den Eingabedialog reserviert.

Nach dem Start des Programms fordert der Computer von uns die Entscheidung, ob wir neue Zahlen eingeben oder gespeicherte von der Diskette lesen wollen. Zunächst neue Zahlen: Nachdem ein Titel gewählt ist, nimmt er bis zu 32 Zahlen an. Als nette Geste druckt er von vornherein eine Null an die Stelle, wo die Eingabe erwartet wird. Psychologisch ausgekocht, wie er ist, rechnet er damit, daß man diese Zahl oft benützen möchte, wo er es einem doch so leicht macht! Aus der Zeit meiner Alkohol-Entwöhnung mittels Computer weiß ich, daß jede Null einer Siegesmeldung gleich kommt. Zur Kontrolle wird nach jeder Eingabe die Summe und der Mittelwert angezeigt. Die Senkung dieses Mittelwertes übt auf den Betroffenen eine ähnliche hypnotische Wirkung aus wie die Steigerung des Tagespensums eines Joggers. Korrektur von falsch eingegebenen Werten ist auch möglich. Der Cursor springt auf das vorhergehende Feld, der Computer löscht die alte Eingabe und korrigiert Summe und Mittelwert. Nach der letzten Zahl kann man die Werte entweder gleich auf Diskette speichern oder sich die Balken zeigen lassen.

Nun zeichnet er Balken für Balken, automatisch das Ganze in die Mittenposition gesetzt, falls es weniger als 32 sind. Der höchste Wert ist Obergrenze, die anderen prozentuale Anteile davon. Selbstverständlich fehlen nicht Titel, Anzahl der Werte, Summe, Mittelwert, Minimum, Maximum und ein Maßstab in Zehnereinteilung. Das sieht schon recht beeindruckend aus. ►

Jetzt geht es aber erst los: Durch Druck auf die Taste »H« für Hilfe wird sie einem zuteil in Form einer Menüliste, die einem verrät, was man mit diesen Werten alles anstellen kann.

Am Beispiel Kurts wollen wir die Möglichkeiten des Programms durchgehen. Eingegeben haben wir die Zigaretten des Monats Dezember 1982. Kurt rauchte zwischen 20 und 30 gewöhnlich. Jetzt möchten wir wissen, wie oft und wann er über 25 Stück — die Menge, ab der ihm meist am nächsten Morgen schlecht ist — geraucht hat. Taste »G« schafft Klarheit. Nach der Eingabe »25« werden alle Werte, die darüber liegen, grafisch anders dargestellt, und wir erkennen, daß er am 4., am 11. und am 31. über seiner Grenze lag. Ein Druck auf die Taste »W«, welcher die Wochentage sichtbar macht, belehrt ihn, daß er an Samstagen besonders viel geraucht hat, außer an dem einen, an dem er auf der Hütte war. Ein »Aha-Erlebnis« für Kurt.

Ein Druck auf die Prozenttaste »Z« bewirkt, daß alle Werte prozentual angezeigt werden, in bezug auf den zuletzt gespeicherten Oberwert 25. Er sieht: Am 4. = 104%, am 11. = 120%, am 31. (Silvester!) = 116%.

Jetzt wollen wir den Trend wissen, das heißt den Mittelwert aller vorangegangenen Werte. Man kann ihn sich vom Rechner mit und ohne Zusatzwerte zeigen lassen. Ohne Zusatzwerte sehen wir, daß er zum Monatsende leicht abgenommen hat. Interessanter wird

```
10 DEF ZF=INT(100*RND-50)
15 CALL CLEAR
20 GOTO 25
25 V$="0050547C38382828"
30 CALL CHAR(143,V$)
35 FOR I=1 TO 16
40 CALL SPRITE(#I,143,I,14*I,16*
I,ZF,ZF)
45 NEXT I
48 CALL KEY(0,K,S):: IF S<1 THEN
 100
50 GOTO 20
99 !BALKENDIAGRAMM  TI99/4-32 WE
RTE*RUEDIGER NEUMANN
100 CALL SCREEN(4):: SP=3
120 DISPLAY AT(14,6):"BALKENDIAG
RAMM 32"
130 FOR CH=120 TO 143 :: DISPLAY
 AT(18,SP):CHR$(CH)::: SP=SP+1 :
: NEXT CH
190 SP=9 :: FOR CH=104 TO 110 ::
 DISPLAY AT(9,SP):CHR$(CH):: SP=
SP+1 :: NEXT CH :: DISPLAY AT(9,
SP+1):CHR$(92)&CHR$(42)
210 R,FF=2 :: G=12 :: KK=13 :: H
=16 :: J=1
220 FOR I=2 TO 4 :: CALL COLOR(I
,R,H):: NEXT I
230 FOR I=5 TO 8 :: CALL COLOR(I
,R,G):: NEXT I
240 CALL COLOR(9,FF,J):: CALL CO
LOR(10,R,H):: CALL COLOR(11,H,J)
250 FOR I=12 TO 14 :: CALL COLOR
(I,R,H):: NEXT I
260 DIM Q(32),A(32),D(32),E(32),
P(32)
270 ! DEFINITION DER BALKEN
272 CALL CHAR(61,"0000003C003C00
00")
275 CALL CHAR(93,"FFFF0000000000
00")
280 CALL CHAR(96,"00000000000000
7C")
290 CALL CHAR(97,"00000000000007C
7C")
300 CALL CHAR(98,"000000000007C7C
7C")
310 CALL CHAR(99,"00000007C7C7C7C
7C")
320 CALL CHAR(100,"0000007C7C7C7
C7C")
330 CALL CHAR(101,"00007C7C7C7C7
C7C")
340 CALL CHAR(102,"007C7C7C7C7C7
C7C")
350 CALL CHAR(103,"7C7C7C7C7C7C7
C7C")
390 CALL CHAR(92,"0000040C14247C
00")!DELTA
400 CALL CHAR(111,"000000FFFF000
000")
410 CALL CHAR(112,"0000000000000
0FF")!LINIE UNTEN
415 CALL CHAR(91,"000000FF000000
00")!WAAGER.LINIE MITTE
425 ! ALLE ZAHLEN VON 0-32
440 CALL CHAR(120,"00005C5454545
45C")!10
450 CALL CHAR(121,"00006C2424242
424")!11
460 CALL CHAR(122,"00005C5444485
05C")!12
470 CALL CHAR(123,"00005C444C445
45C")!13
480 CALL CHAR(124,"0000D050545E4
444")!14
490 CALL CHAR(125,"00005C505C445
45C")!15
500 CALL CHAR(126,"00005C505C545
45C")!16
510 CALL CHAR(127,"00005C4448584
848")!17
520 CALL CHAR(128,"00005C545C545
45C")!18
530 CALL CHAR(129,"00005C545C445
45C")!19
540 CALL CHAR(130,"00007C3434745
47C")!20
560 CALL CHAR(131,"0000741434644
474")!21
570 CALL CHAR(132,"00006C2424484
86C")!22
580 CALL CHAR(133,"00006C242C446
C6C")!23
590 CALL CHAR(134,"000068282C444
464")!24
600 CALL CHAR(135,"00006C282C444
46C")!25
610 CALL CHAR(136,"00006C28284C4
C6C")!26
620 CALL CHAR(137,"00006C242C444
464")!27
630 CALL CHAR(138,"00007C343C545
47C")!28
640 CALL CHAR(139,"00007C541C244
46C")!29
650 CALL CHAR(140,"00006C2C6C2C2
C6C")!30
660 CALL CHAR(141,"00006C2464242
464")!31
670 CALL CHAR(142,"00006C2464282
86C")!32
675 CALL CHAR(143,"007C443060447
C00")!SUMME
680 CALL CHAR(113,"040404043C040
404")
690 CALL CHAR(114,"0000001C04040
404")
700 CALL CHAR(115,"040404041C040
404")
710 CALL CHAR(116,"0404040404040
41C")
720 CALL CHAR(117,"0404040404040
404")
730 CALL CHAR(118,"041C040404040
404")
740 CALL CHAR(119,"0404041C04040
404")
770 ! ---  WOCHENTAGE +UMLAUTE -
------
775 CALL CHAR(104,"0070407C14547
C00")!SO
780 CALL CHAR(105,"0050707C54545
C00")!MO
790 CALL CHAR(106,"0070545054547
400")!DI
800 CALL CHAR(107,"0050747054545
400")!MI
810 CALL CHAR(108,"0060505C54546
C00")!DO
820 CALL CHAR(109,"0070406C48484
A00")!FR
830 CALL CHAR(110,"0060406C242C6
C00")!SA
835 CALL CHAR(36,"00443844444438
00")!UE
840 CALL CHAR(64,"00441028447C44
00")!AE
845 CALL CHAR(95,"00440044444430
00")!OE
900 D(0),MIW=0 :: CALL HCHAR(21,
3,112,28)
910 DISPLAY AT(23,1)BEEP:"NEUE W
ERTE EINGEBEN        =0" :: DISPL
AY AT(24,1):"GESPEICHERTE WERTE
ABRUFEN=1"
920 CALL KEY(0,K,S):: IF S=0 THE
N 920
 922 CALL DELSPRITE(ALL)
925 CALL CLEAR
930 IF K=48 THEN 940 ELSE IF K=4
9 THEN 13000
940 CALL CLEAR :: DISPLAY AT(11,
11):"TITEL:" :: CALL HCHAR(14,3,
91,28)
945 FOR I=1 TO 32 :: A(I),E(I)=0
 :: NEXT I :: SU,SUM,US,MI=0
950 DISPLAY AT(13,1):TITEL$ :: A
CCEPT AT(13,1)BEEP SIZE(-28):TIT
EL$
1000 CALL CLEAR !WERTE-EINGABE
1010 Z3=4 :: S3=1
1020 CALL HCHAR(2,1,91,32):: CAL
L HCHAR(21,1,91,32)
1030 CALL HCHAR(23,1,111,32):: D
ISPLAY AT(24,1):"FEHLER=E+ENT  E
NDE =  .  "
1040 TA1=INT((28-LEN(TITEL$))/2+
.5)
1050 DISPLAY AT(1,TA1):TITEL$
1060 B=1
1070 IF B=17 THEN Z3=4 :: S3=15
:: CALL VCHAR(4,16,117,16)
1080 DISPLAY AT(Z3,S3):USING 602
0:B
1090 ACCEPT AT(Z3,S3+8)SIZE(-6)V
ALIDATE(DIGIT,".E")BEEP:A2$
1100 IF A2$="E" THEN GOSUB 1200
1110 IF A2$="." THEN DISPLAY AT(
Z3,S3):"" :: GOTO 1160
1120 A(B),E(B)=VAL(A2$)
1130 SU=SU+A(B):: MI=SU/(B+SUM):
: Z3=Z3+1
1140 DISPLAY AT(22,3)SIZE(-11):U
```

dieses Verfahren zusammen mit den Werten eines oder mehrerer Vormonate. Der Computer fragt nach der Summe der alten Werte und ihrer Anzahl. Kurt könnte damit den Zigarettenkonsum seit seiner Geburt verfolgen und Schlüsse daraus ziehen. Das Trend-Diagramm sieht beispielsweise für den Monat Januar ausgesprochen freundlich aus. Da hatte er ja das Rauchen »für immer« aufgegeben ... Den schweren Rückfall am 15., der im Originalbild wie ein Wolkenkratzer wirkt, nimmt man im Trenddiagramm gerade noch als Ausrutscher wahr.

Wer jetzt noch nicht genug hat, kann über die Taste »K« im Diagramm Werte verändern oder hinzufügen. Auch Wegnahme ist möglich, wie wir später sehen. (Wie hätte mein Mittelwert ausgesehen, wenn ich am 15. nicht geraucht hätte?) Man kann dieses Illusionsspiel so lange fortführen, bis nichts mehr an das ursprüngliche Diagramm erinnert.

Wenn man dann keine Ahnung mehr hat, wie die alte Lage war, genügt ein Druck auf »A«, und die ehemaligen Verhältnisse sind wiederhergestellt, der Computer kehrt zum Ausgangsbild zurück. (Für Insider: Alle Werte werden von vornherein mit zwei verschiedenen Variablen belegt; mit der einen kann man »spielen«, die andere bleibt jungfräulich, bis man ein neues Programm eingibt.)

Die anderen Wahlmöglichkeiten: »S« sortiert rauf und runter.

```
SING 6012:SU :: DISPLAY AT(22,15
):USING 6013:MI
1150 B=B+1 :: IF B<33 THEN 1070
1160 B,B2=B-1 :: DISPLAY AT(24,1
):"ABSPEICHERN=B     WEITER= ENT"
1170 CALL KEY(0,K,S):: IF S=0 TH
EN 1170
1180 IF K=66 THEN GOSUB 14000 !A
BSPEICHERN
1190 GOTO 1400
1200 B=B-1 :: SU=SU-A(B):: MI=SU
/(B-1)
1210 DISPLAY AT(22,3)SIZE(-11):U
SING 6012:SU :: DISPLAY AT(22,15
):USING 6013:MI
1220 DISPLAY AT(Z3-1,S3):USING 6
020:B
1230 ACCEPT AT(Z3-1,S3+8)SIZE(-6
)VALIDATE(DIGIT,".E"):A2$ :: IF
A2$="E" THEN Z3=Z3-1 :: GOTO 120
0
1240 A(B),E(B)=VAL(A2$)
1250 Z3=Z3-1
1260 RETURN
1400 SUMM=SU :: MIT=MI :: IF SU=
0 THEN DISPLAY AT(2,1)BEEP:"KEIN
 PROGRAMM GELADEN!" :: GOTO 900
1410 ANF,FAN=INT((31-B)/2+.5)
1420 WO=8
1500 TIT$=""
1520 CALL CLEAR
1740 GOTO 2050
1760 DISPLAY AT(24,1):"ICH SORTI
ERE AUFW@RTS- WARTE"
1765 FF=5
1770 TIT$="(AUFW@RTS SORTIERT)"
1790 SH=0
1800 FOR I=1 TO B-1
1810 IF A(I+1)>=A(I)THEN 1860
1820 HS=A(I)
1830 A(I)=A(I+1)
1840 A(I+1)=HS
1850 SH=1
1860 NEXT I
1870 IF SH>0 THEN 1790
1900 GOTO 2050
1910 DISPLAY AT(24,1):"ICH SORTI
ERE ABW@RTS- WARTE!"
1930 FF=14
1940 TIT$="(ABW@RTS SORTIERT)"
1970 SH=0 :: FOR I=1 TO B-1 :: I
F A(I+1)<=A(I)THEN 2020 :: HS=A(
I):: A(I)=A(I+1):: A(I+1)=HS ::
SH=1
2020 NEXT I
2030 IF SH>0 THEN 1970
2050 ! MAX&MIN
2060 C,MA,IM=A(1):: FOR I=1 TO B
 :: C,MA=MAX(A(I),C):: IM=MIN(A(
I),IM):: NEXT I
2155 IF AF=0 THEN AF=1
2220 ! UMRECHNUNG IN PROZENTE
2230 FOR I=1 TO B :: P(I)=A(I)/C
*100 :: P(I)=INT(P(I)+.5):: NEXT
 I
2270 ! VORBEREITUNG DES BILDES
2285 TA1=INT((28-LEN(TITEL$))/2+
.5)
2288 TA2=INT((28-LEN(TIT$))/2+.5
)
2290 TA3=INT((28-LEN(TI$))/2+.5)
2300 DISPLAY AT(1,TA1):TITEL$ ::
 DISPLAY AT(2,1):"" :: DISPLAY A
T(2,TA2):TIT$
2310 DISPLAY AT(3,TA3):TI$ :: IF
 TI$="" THEN CALL HCHAR(3,1,91,3
2)
2510 DISPLAY AT(3,29-LEN(STR$(C)
))SIZE(-LEN(STR$(C+1))):STR$(C)
2520 CALL HCHAR(18,1,91,32)
2522 CALL HCHAR(20,1,111,32)
2530 ! ZEICHNEN DER BALKEN
2540 CH=AF+48 :: IF AF>9 THEN CH
=AF+110
2550 WT=104+WO
2556 FAN=ANF :: CALL HCHAR(16,1,
112,FAN):: IF FAN+B+1>32 THEN 25
57 :: CALL HCHAR(16,FAN+B+1,112,
32-FAN-B)
2557 US,MI=0
2560 FOR I=1 TO B
2562 US=US+A(I):: MI=US/I
2570 X=I+((32-B)/2)-.5
2580 IF B<30 THEN 2600
2590 X=I
2600 IF ANF=0 THEN 2620
2610 X=I+ANF-1
2620 AD=INT(P(I)/8)
2630 IF P(I)<=100 THEN 2665
2650 CALL VCHAR(4,X,101,13)!UEBE
RLAUF
2660 GOTO 2720
2665 CALL VCHAR(4,X,32,13)
2670 CALL VCHAR(17-AD,X,103,AD)
2690 BE=P(I)-AD*8
2700 IF BE=0 THEN 2720
2710 CALL HCHAR(16-AD,X,95+BE)
2720 CALL HCHAR(17,X,CH)
2730 IF WO=8 THEN 2780
2740 CALL HCHAR(19,X,WT)
2750 WT=WT+1
2760 IF WT>110 THEN 2770 ELSE 27
80
2770 WT=104
2780 CH=CH+1
2790 IF CH=58 THEN CH=120
2810 NEXT I
2812 DISPLAY AT(21,6):B;"WERTE:"
;STR$(IM);"-";STR$(MA)
2814 DISPLAY AT(22,7):CHR$(143);
"=";STR$(US);" MI=";STR$(INT(MI*
100+.5)/100)
2820 IF B=32 THEN 2990
2830 IF B+ANF>32 OR B+ANF<2 THEN
 2990
2850 W=X+1
2855 !MASSSTAB:
2860 CALL VCHAR(4,W,114)
2870 CALL VCHAR(5,W,115)
2880 CALL VCHAR(6,W,116)
2890 CALL VCHAR(7,W,117)
2900 CALL VCHAR(8,W,118)
2910 CALL VCHAR(9,W,119)
2920 CALL VCHAR(10,W,113)
2930 CALL VCHAR(11,W,116)
2940 CALL VCHAR(12,W,117)
2950 CALL VCHAR(13,W,118)
2960 CALL VCHAR(14,W,119)
2970 CALL VCHAR(15,W,115)
2980 CALL VCHAR(16,W,116)
2990 CALL HCHAR(23,1,91,32):: DI
SPLAY AT(24,1):"
   H=HILFE"
3000 CALL SOUND(200,368,0,369,2,
878,5)
3010 CALL KEY(0,K,S):: IF S=0 TH
EN 3010
3014 IF K=80 THEN CALL COPY
3015 IF K=80 THEN 2220
3016 IF K=72 THEN 5000 !HELP
3020 IF K<>65 THEN 3022
3021 FOR L=1 TO 32 :: CALL VCHAR
(4,L,32,16):: NEXT L :: TIT$=""
:: FF=2 :: AF=1 :: J=1 :: D(0),M
IW=0 :: GOTO 4140 !ANFANGSBEISPI
EL
3022 IF K=90 THEN 3130 !PROZENTW
ERTE
3023 IF K=66 THEN GOSUB 14000
3024 IF K=87 THEN 8000 !WOCH
3025 IF K=83 THEN 3460 !SORTIER
3026 IF K=68 THEN 3590 !DELTA-%
3028 IF K=50 THEN 3240 !2 GW
3029 IF K=89 THEN AF=1 :: GOTO 1
3000 !DATEI
3030 IF K=71 THEN FF=5 :: CALL S
CREEN(9):: DISPLAY AT(24,1):"WEL
CHE OBERGRENZE ?";C :: ACCEPT AT
(24,21)VALIDATE(DIGIT,".")SIZE(-
8):C :: GOTO 2220 !ANDERER OBERW
ERT
3031 IF K=76 THEN 3190 !LESEN
3032 IF K=84 THEN 3870 !TREND
3033 IF K=78 THEN 900 !NEUESBEIS
P.
3034 IF K=33 THEN STOP
3035 IF K=70 THEN 3036 ELSE 3038
 !AND.POS.
3036 FOR I=1 TO 32 :: CALL VCHAR
(4,I,32,16):: NEXT I :: DISPLAY
AT(24,1):"NEUE   POSITION?";33-B
3037 ACCEPT AT(24,18)SIZE(-2)VAL
IDATE(DIGIT):ANF :: IF ANF+B>33
THEN 3037 :: GOTO 2220
3038 IF K=67 THEN DISPLAY AT(24,
1):"NEUE ANFANGSZAHL ";33-B :: A
CCEPT AT(24,19)VALIDATE(DIGIT)SI
ZE(-2):AF :: IF AF>33-B THEN 303
8 :: GOTO 2050
3040 IF K<>75 THEN 2220 !GEHABTE
S BILD
3043 IF K=75 THEN DISPLAY AT(24,
1):"WELCHER WERT? ";B+1 :: ACCEP
T AT(24,16)VALIDATE(DIGIT)SIZE(-
2):ZW
```

»L« = Lesen. Alle eingegebenen beziehungsweise durch Rechenoperationen ermittelten Werte werden angezeigt.

»F« und »C« = Neue Anfangspositionen. Wenn eine Analyse mitten im Monat beginnt, berücksichtigt der Computer auch dies zur besseren Veranschaulichung.

»2« = Zwei Grenzwerte setzen. Alle Werte, die unter dem ersten und über dem zweiten abgefragten Wert liegen, werden aufgelistet.

»D« = Delta-Prozentwerte. Der Unterschied eines jeden Wertes vom vorhergegangenen wird in Prozenten berechnet.

»P« = Druckerprogramm. Der jeweilige Bildschirminhalt wird, außer der letzten Zeile, die ja nur Dialoginhalte führt, auf dem Thermoprinter, so angeschlossen, ausgedruckt. Wer Einzelzeilen drucken will, muß die Copy-Subroutine ändern. Ein entsprechendes Programm findet sich in CP 1/83, Seite 44. Auch das Menü kann ausgedruckt werden.

»N« = Neues Beispiel. Eine neue Wertetabelle kann eingegeben werden; die alten Zahlen werden, wenn man sie jetzt noch nicht gespeichert hat, »vernichtet«, außer man geht in die Dateiverarbeitung, dann bleiben Werte, die größer sind als die alten, noch erhalten. Dies ist praktisch, wenn man einen halben Monat mit den Zahlen eines anderen »ausfüllen« möchte.

Nun zur Dateiverarbeitung: Gleich am Anfang bietet einem der Rechner auf die Entscheidung zur

```
3044 IF ZW>B+1 THEN 3043 :: IF Z
W=B+1 THEN B=ZW
3045 DISPLAY AT(24,1):"NEUER WER
T? "&A(ZW):: ACCEPT AT(24,14)SIZ
E(-8)VALIDATE(DIGIT,"."):NW :: A
(ZW)=NW :: GOTO 2050
3130 TI$="PROZENTWERTE" :: GOSUB
 6000
3135 FA=AF
3138 FOR I=1 TO B
3139 IF I=INT(B/2)+1 THEN Z3=4 :
: S3=15
3140 DISPLAY AT(Z3,S3):USING "##
:":FA :: DISPLAY AT(Z3,S3+3):USI
NG "###.#=":A(I):: DISPLAY AT(Z3
,S3+9):USING "###.#":A(I)/C*100
:: Z3=Z3+1 :: FA=FA+1
3170 NEXT I
3173 !CALL VCHAR(4,16,117,B/2)
3175 TIT$,TI$=""
3180 GOTO 2990
3190 GOSUB 6000 !LESEN
3192 FA=AF
3195 FOR I=1 TO B
3196 SU=SU+A(I):: MI=SU/I
3197 IF I=INT(B/2)+1 THEN Z3=4 :
: S3=15
3200 DISPLAY AT(Z3,S3):USING 602
2:FA :: DISPLAY AT(Z3,S3+6):USIN
G 6040:A(I):: Z3=Z3+1 :: FA=FA+1
3230 NEXT I
3231 DISPLAY AT(21,6):B;"WERTE:"
;STR$(IM);"-";STR$(MA)
3232 DISPLAY AT(22,7):CHR$(143);
"=";STR$(SU);" MI= ";STR$(INT(MI
*100+.5)/100)
3233 CALL VCHAR(4,16,117,B/2)
3234 GOTO 2990
3240 !2 GRENZWERTE
3245 Z3=4 :: S3=1
3250 FA=AF :: DISPLAY AT(24,1):"
UNTERER GRENZWERT:";IM :: ACCEPT
 AT(24,20)SIZE(-8)VALIDATE(DIGIT
,".")BEEP:GW1$
3260 IF GW1$="" THEN 3280
3270 GW1=VAL(GW1$)
3280 DISPLAY AT(24,1):"OBERER  G
RENZWERT:";MA :: ACCEPT AT(24,20
)SIZE(-8)VALIDATE(DIGIT,".")BEEP
:GW2$
3290 IF GW2$="" THEN 3320
3300 GW2=VAL(GW2$)
3310 IF GW2<=GW1 THEN 3280
3315 FOR L=1 TO 32 :: CALL VCHAR
(4,L,32,16):: NEXT L
3320 FOR I=1 TO B
3325 IF I=16 THEN Z3=4 :: S3=15
3330 IF GW1$="" THEN 3350
3340 IF A(I)<=GW1 THEN 3375
3350 IF GW2$="" THEN 3380
3360 IF A(I)<GW2 THEN 3380
3375 DISPLAY AT(Z3,S3):USING 602
2:FA :: DISPLAY AT(Z3,S3+5):USIN
G 6040:A(I):: Z3=Z3+1
3380 FA=AF+I :: NEXT I
3390 GOTO 2990
3460 DISPLAY AT(24,1):"SORTIEREN
:  AUF=1  AB=2"
3470 CALL KEY(0,K,S):: IF S=0 TH
EN 3470
3490 IF K=49 THEN 1760 ELSE IF K
=50 THEN 1910 :: GOTO 3470
3590 TI$=CHR$(92)&"-PROZENTE" !D
ELTA-PROZENT
3595 GOSUB 6000
3600 DISPLAY AT(4,1)SIZE(-13):"K
EIN1. WERT !"
3602 FA=AF+1
3603 FOR I=2 TO B :: IF A(I-1)<>
0 THEN DPR=(A(I)-A(I-1))/A(I-1)*
100 ELSE IF A(I-1)=0 THEN 3690
3605 IF I=INT(B/2)+1 THEN Z3=3 :
: S3=15
3630 DISPLAY AT(Z3+1,S3):USING 6
022:FA :: DISPLAY AT(Z3+1,S3+6):
USING 6040:DPR :: Z3=Z3+1 :: FA=
FA+1
3670 NEXT I
3675 CALL VCHAR(4,16,117,B/2)
3680 TI$="" :: GOTO 2990
3690 DISPLAY AT(Z3+1,S3):USING 6
022:FA :: DISPLAY AT(Z3+1,S3+7):
"UNENDL" :: Z3=Z3+1 :: FA=FA+1
3700 GOTO 3670
3870 !  TRENDANALYSE
3880 FF=15 :: CALL SCREEN(6)
3900 D(0),MIW=0 :: DISPLAY AT(24
,1)BEEP:"MIT ZUSATZWERTEN=1 OHNE
=0 "
3910 SU=0 :: NI=1
3930 CALL KEY(0,K,S):: IF S=0 TH
EN 3930
3960 IF K=49 THEN 3980 ELSE IF K
=48 THEN 4030
3970 GOTO 3900
3980 DISPLAY AT(24,1):"" :: DISP
LAY AT(24,1):"ALTE ";CHR$(143):
3990 ACCEPT AT(24,8)SIZE(-9)VALI
DATE(NUMERIC):D(0)
4000 SU=0
4010 DISPLAY AT(24,1):"WIEVIEL W
ERTE VORHANDEN? " :: ACCEPT AT(2
4,26)VALIDATE(DIGIT):NI
4020 NI=NI+1
4030 SU=0
4060 FOR I=1 TO B :: D(I)=A(I)+D
(I-1):: A(I)=INT(D(I)/NI*100+.5)
/100 :: SU=SU+A(I):: NI=NI+1
4090 NEXT I
4100 MA,IM=A(1):: FOR I=1 TO B :
: MA=MAX(A(I),MA):: IM=MIN(A(I),
IM):: NEXT I
4110 BB=NI-1 :: MIW=INT((SU/BB)*
100+.5)/100
4120 TIT$="TR:N="&STR$(BB)&" "&C
HR$(143)&"="&STR$(SUMM+D(0))&" M
I="&STR$(INT((SUMM+D(0))/BB*100+
.5)/100)
4130 GOTO 2220
4140 ! ALTE WERTE
4145 IF ZW=B THEN B=B2
4150 FOR I=1 TO B
4160 A(I)=0 :: A(I)=E(I)
4180 SU=SUMM :: MI=MIT :: WO=8
4200 NEXT I
4205 ANF,FAN=INT((31-B)/2+.5)::
CALL SCREEN(3)
4210 GOTO 2050
5000 !HELP
5010 DISPLAY AT(1,11)ERASE ALL:"
IHRE WAHL"
5015 CALL HCHAR(2,13,111,9)
5020 DISPLAY AT(4,3):"WERTE LESE
N          = L"
5030 DISPLAY AT(5,3):"WERTE SORT
IEREN      = S"
5035 DISPLAY AT(6,3):"ANDERE OBE
RGRENZE    = G"
5037 DISPLAY AT(7,3):"NEUE ANF.P
OSITION    = F"
5040 DISPLAY AT(8,3):"TRENDANALY
SE          = T"
5050 DISPLAY AT(9,3):"ZWEI GRENZ
WERTE      = Z"
5055 DISPLAY AT(10,3):"PROZENTWE
RTE         = Z"
5060 DISPLAY AT(11,3):CHR$(92);"
% -WERTE              = D"
5062 DISPLAY AT(12,3):"NEUE ANFA
NGSZAHL    = C"
5065 DISPLAY AT(13,3):"WERTE AUS
DRUCKEN    = P"
5070 DISPLAY AT(15,3):"WERTE ABS
PEICHERN   = B"
5080 DISPLAY AT(16,3):"DATEIVERA
RBEITUNG   = Y"
5083 DISPLAY AT(17,3):"MIT WOCHE
NTAGEN     = W"
5084 DISPLAY AT(18,3):"ANF_GEN+K
ORRIGIEREN= K"
5085 DISPLAY AT(19,3):"ANFANGSBE
ISPIEL     = A"
5090 DISPLAY AT(20,3):"NEUES BEI
SPIEL      = N"
5100 DISPLAY AT(21,3):"PROGRAMME
NDE          = !"
5105 CALL KEY(0,K,S):: IF S=0 TH
EN 5105 :: IF K<>80 THEN CALL CL
EAR
5110 GOTO 3014
6000 !LESEVORBEREITUNG
6005 SU,MI=0
6010 Z3=4 :: S3=1
6012 IMAGE "SU=####.##"
6013 IMAGE "MI =###.##"
6020 IMAGE "NR:## = 0"
6022 IMAGE "NR:##="
6025 TA2=INT((28-LEN(TIT$))/2+.5
)
6027 TA3=INT((28-LEN(TI$))/2+.5)
6030 DISPLAY AT(1,TA1):TITEL$ ::
 DISPLAY AT(2,1):"" :: DISPLAY A
T(2,TA2):TIT$
6035 DISPLAY AT(3,TA3):TI$ :: IF
  TI$="" THEN CALL HCHAR(3,1,91,3
```

Problemlösungen

Dateiverarbeitung mit Disketten auf Wunsch einen Katalog aller auf Diskette gespeicherten Daten. Dieser Katalog kann auch gedruckt werden. Jetzt wird ein Programm geladen. Bevor man zum Grafikdiagramm geht, kann man entscheiden, ob man sämtliche Werte dazu hernehmen will oder eine Auswahl treffen möchte. Der Computer verarbeitet dann nur die ausgewählten Werte, hat aber alle anderen im »Gedächtnis« und kann sie einzeln mit der »K«-Taste abrufen.

Das Abspeichern in die Datei geht problemlos, sowohl nach der Eingabe als auch aus dem Diagramm, auch von den veränderten oder aufgefüllten Datensätzen.

Zeile 9000: Sie wird im Laufe des Programms nicht angesprungen. Hier sind alle codierten Sonderzeichen abrufbar, denn die Zahlen über 9 mußten ja, da sie mehr als einen Spaltenplatz erfordert hätten, neu geschaffen werden.

Wer andere Sonderzeichen braucht, kann die Zeilennummer wegeditieren, dann druckt der Computer die Codes einzeln aus.

Die mathematischen Transaktionen und Programmkünste haben mich über ein Jahr beschäftigt. Alle Zahlenspiele sind an Beispielen nachgerechnet worden. Da ich aber weder in der Mathematik noch beim Hobby-Programmieren den Anspruch der Unfehlbarkeit erhebe, bin ich über einen Hinweis auf Erleichterungen oder Verbesserungen nicht erbost.

(Rüdiger Neumann)

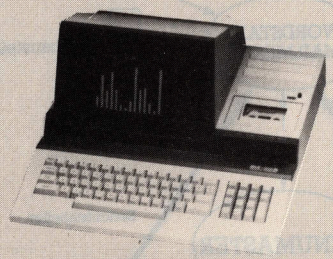
```
2)
6040 IMAGE "####.##"
6050 RETURN
8000 !WOCHENTAGE+MAGNIFY
8080 CALL SPRITE(#1,104,5,92,45)
8090 CALL SPRITE(#2,105,13,78,70
)
8100 CALL SPRITE(#3,106,2,92,96)
8110 CALL SPRITE(#4,107,14,78,11
4)
8120 CALL SPRITE(#5,108,7,92,138
)
8130 CALL SPRITE(#6,109,11,78,16
2)
8140 CALL SPRITE(#7,110,16,92,18
6)
8150 CALL MAGNIFY(2)
8210 DISPLAY AT(24,1):"SO=OMO=1O
I=2MI=3OO=4FR=5SA=6"
8230 ACCEPT AT(24,28)SIZE(-1)VAL
IDATE("O1234568"):WO
8235 CALL DELSPRITE(ALL)
8240 GOTO 2220
9000 OPEN #1:"TP.E.U",OUTPUT ::
FOR S1=91 TO 143 :: CALL CHARPAT
(S1,F1$):: PRINT #1:S1;CHR$(S1);
"=";F1$ :: NEXT S1
12000 FOR S=1 TO X-1 :: CALL HCH
AR(17,S,112):: NEXT S
12500 !KATALOG
12505 CALL CLEAR :: OPEN #1:"DSK
1.",INPUT ,RELATIVE,INTERNAL
12510 INPUT #1:TITEL$,J,J,K
12517 PRINT
12520 DISPLAY "DATEINAME"
12525 PRINT RPT$(CHR$(91),28)
12530 FOR SCHL=1 TO 127
12540 INPUT #1:D$,A1,J,K
12550 IF D$="" THEN 12630
12560 IF ABS(A1)<>3 THEN 12620
12570 IF K<>80 THEN 12620
12580 DISPLAY D$,
12590 IF A1>0 THEN 12610
12600 PRINT TAB(13);"J";
12610 !PRINT RPT$("-",10)
12620 NEXT SCHL
12630 CLOSE #1
12635 PRINT RPT$(CHR$(91),20)
12636 DISPLAY AT(24,1)BEEP:"PRIN
T=P        WEITER=ENTER"
12637 CALL KEY(0,K,S):: IF S=0 T
HEN 12637
12638 IF K=80 THEN CALL COPY
12640 GOTO 13070
13000 !DATEIVERARBEITUNG
13005 D1=1 :: DISPLAY AT(24,1)BE
EP:"KATALOG=K    WEITER= 0"
13010 CALL KEY(0,K,S):: IF S=0 T
HEN 13010
13020 IF K=75 THEN 12500 ELSE IF
 K=48 THEN 13070
13030 GOTO 13000
13070 DISPLAY AT(24,1)BEEP:"TITE
L ODER BLANK>";TITEL$ :: ACCEPT
AT(24,18)SIZE(-10):D$ :: TITEL$=
D$
13080 IF D$="" THEN 13230
13090 OPEN #10:"DSK1."&TITEL$,RE
LATIVE 32,INTERNAL
13092 ON ERROR 5000
13100 ND=0 :: DISPLAY AT(24,1)BE
EP:"BALKEN=1  LESEN+AUSWAHL=ENT"
13110 CALL KEY(0,KEY,S):: IF S=0
 THEN 13110
13120 IF KEY=49 THEN ND=1
13150 GOTO 13470
13230 CALL CLEAR :: !MENUE
13300 DISPLAY AT(14,3):"N= NEUES
 PROGRAMM"
13303 DISPLAY AT(16,3):"K= KATAL
OG"
13305 DISPLAY AT(18,3):"P= BALKE
N"
13307 DISPLAY AT(20,3):"B= NEUAN
FANG"
13320 DISPLAY AT(24,1):"IHRE WAH
L:"
13330 CALL KEY(0,K,S):: IF S=0 T
HEN 13330
13353 IF K=75 THEN 12500
13355 IF K=80 THEN 1400
13365 IF K=66 THEN 900
13370 IF K=78 THEN 13070
13380 GOTO 13230
13470 B=1
13480 RESTORE #10
13490 IF EOF(10)THEN 13511
13500 INPUT #10:Q(B)
13510 B=B+1 :: GOTO 13490
13511 :: B,B2=B-1 :: SU,MI=0
13513 FAN=INT((32-B)/2+.5):: IF
B>27 THEN ANF,FAN=1
13514 FOR I=1 TO B :: A(I),E(I)=
Q(I):: SU=SU+A(I):: MI=SU/(I)::
NEXT I
13515 IF SU=0 THEN DISPLAY AT(3,
4)BEEP:TITEL$;" GIBT ES NICHT!"
:: CLOSE #10:DELETE :: GOTO 1300
0
13518 IF ND=1 THEN CLOSE #10 ::
GOTO 1400
13520 TA1=INT((28-LEN(TITEL$))/2
+.5)
13523 DISPLAY AT(1,TA1)ERASE ALL
:TITEL$
13525 CALL HCHAR(23,1,91,32):: C
ALL HCHAR(3,1,91,32)
13526 FA=D1 :: IF D1=0 THEN FA=1
13527 :: Z3=5 :: S3=1 :: SU,MI=0
13528 FOR I=1 TO B
13529 IF I=INT(B/2)+1 THEN Z3=5
:: S3=15 :: CALL VCHAR(5,16,117,
B/2)
13530 DISPLAY AT(Z3,S3):USING 60
22:FA
13550 DISPLAY AT(Z3,S3+6):USING
6040:A(I)
13555 SU=SU+A(I):: MI=SU/(I+SUM)
:: Z3=Z3+1
13557 DISPLAY AT(22,3)SIZE(-11):
USING 6012:SU :: DISPLAY AT(22,1
5):USING 6013:MI
13560 FA=FA+1 :: NEXT I
13564 DISPLAY AT(24,1):"BALKEN=B
   AUSWAHL=A   /ENT"
13565 CALL KEY(0,K,S):: IF S=0 T
HEN 13565
13566 IF K=65 THEN 13600
13570 IF K=66 THEN CLOSE #10 ::
GOTO 1400
13580 CLOSE #10 :: GOTO 13230
13600 DISPLAY AT(24,1):"AB SATZ:
 1" :: ACCEPT AT(24,10)VALIDATE(
DIGIT)SIZE(-2):D1
13605 DISPLAY AT(24,13):"BIS SAT
Z:";B :: ACCEPT AT(24,23)VALIDAT
E(DIGIT)SIZE(-2):D2
13700 RESTORE #10,REC D1-1
13705 FOR B=1 TO D2-D1+1
13708 IF EOF(10)THEN 13511
13710 INPUT #10:Q(B)
13720 NEXT B
13730 GOTO 13511
14000 !ABSPEICHERN
14010 DISPLAY AT(24,1):"TITEL
  ENT=< ";TITEL$ :: ACCEPT AT(24
,17)SIZE(-10):T$
14020 IF T$="" THEN RETURN
14030 OPEN #11:"DSK1."&T$,RELATI
VE 32,INTERNAL
14040 FOR I=0 TO B-1
14050 Q(I)=A(I+1)
14060 PRINT #11,REC I:Q(I)
14070 NEXT I
14080 CLOSE #11
14090 RETURN
32702 SUB COPY
32704 OPEN #9:"TP.E.U.S",OUTPUT
32705 DISPLAY AT(24,1):"ABBRUCH=
IRGENDEINE TASTE"
32736 FOR Z=1 TO 23 :: IG$="" ::
 FOR S=1 TO 32 :: CALL GCHAR(Z,S
,IV)
32738 I$=CHR$(IV):: IG$=IG$&I$
32740 NEXT S
32743 PRINT #9:TAB(SA);IG$
32744 CALL KEY(0,K,S):: IF S=1 T
HEN 32748
32746 NEXT Z
32748 CLOSE #9
32750 SUBEND
```

Listing des Statistik-Programms von R. Neumann. Die horizontalen Striche im Listing dienen der besseren Unterscheidung der einzelnen Programmteile

Weitere Computer-Knüller am Dontenwill-Stand Hobbytronic '83 Dortmund 27. 4.–1. 5. Westfalenhalle

Heiße Preise für Computer

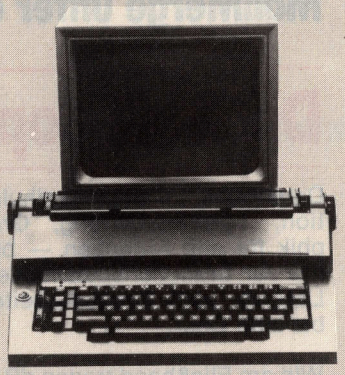
Die ganze Welt der Computer vom Taschenrechner bis zum Mehrplatz-Business-System zu Tiefpreisen. Beispielsweise:



Sharp MZ-80 A
Riesenleistung, kompakt verpackt
Kompaktes, leistungsfähiges Tischcomputer-System mit integriertem Bildschirm (23 cm, 25 x 40 Zeichen) und Kassettenlaufwerk. Bei Dontenwill serienmäßig mit 48 000 Zeichen Speicherkapazität – ermöglicht Direktzugriff auf gespeicherte Daten! Programmiersprachen: BASIC (serienmäßig), PASCAL, ASSEMBLER, Maschinensprache. Ideal für Anfänger, aber auch unbegrenzte Möglichkeiten durch Systemerweiterung. Vielseitig ausbaufähig mit Druckern, Floppys, Terminals usw.
Dontenwill-Preis nur DM 1998,–

Texas TI-99/4A
Der vielseitige Home-Computer
Leistungsstarker Computer für Beruf und Hobby, anschlußfertig für Farb- oder SW-Fernseher. Speicher 16 000 Zeichen, erweiterbar auf 48 000 Zeichen. Viele gebrauchsfertige Programme auf Steckmoduln, z. B. Analyse, Statistik, Datei, Spiele, Computergrafik. BASIC-programmierbar. Einfach und komfortabel zu bedienen. Mit ausführlichem Lehrbuch. Interessante Ausbaumöglichkeiten!
Dontenwill-Preis nur DM 598,– !!!
NEU: PARSEC, das atemberaubende Computerspiel direkt aus dem Weltraum auf Ihren TI-99! Unglaubliche Effekte. PARSEC-Modul zum TI-99/4A **nur DM 98,–**

Sharp PC-1500
Computer-Power im Taschenformat
Superkompakter Computer, der leistungsmäßig fast mit den »Großen« konkurriert. Nur 20 x 11,5 cm klein, überall dabei durch Akku-Betrieb. Speicher 3500 Zeichen (mit Erweiterungsmodul 11 500 Zeichen). 25stellige LCD-Anzeige. Mit Programmierkurs und Programmsammlung (Mathematik, Statistik, E-Technik, Spiele usw.).
Dontenwill-Preis nur DM 498,–
CE-150 Farbgrafik-Drucker dazu (s. Abb.) 4 Farben, 9 Schriftgrößen, Normalpapier. Eingebaute Schnittstelle zum Abspeichern von Daten und Programmen auf Recorder. Komplett mit Akku und Ladegerät **nur DM 398,–**



Daisy 35
Elektron. Speicherschreibmaschine, Schönschreibdrucker, Textsystem
Unschlagbar im Preis-/Leistungsverhältnis! Speicher 5000 Zeichen, aufteilbar in 10 Einzelspeicher. Korrekturspeicher. Anschluß für Recorder als Externspeicher. Typenrad schreibt vor- und rückwärts bis 12 Zeichen/Sek. Viele Automatikfunktionen (Blocksatz, Dauerfunktion usw.). Zahlreiche Editiermöglichkeiten. Suchlauf. Alle Funktionen durch 16-bit-Prozessor vollelektronisch gesteuert.
Dontenwill-Preis nur DM 2198,–
Mit Interface zum Anschluß an Computer + DM 300,–
Komplett mit Bildschirm (siehe Abbildung) **nur DM 3398,–**

Preistelegramm:

NEU: Texas CC-40 nur 678,– !! ; Sharp PC-1212 249,–; Sharp PC-1251 338,–; Sharp CE-125 398,–; Casio FX-602P 239,–; Casio FX-700P 249,–; Texas TI-59 298,–; Texas PC-100C 548,–; Texas TI-57LCD 89,–; Texas TI-55-II 105,–; Texas TI-54 89,–; TI-Programmer 198,–; HP-41C 548,–; HP-41CV 758,–; HP-Drucker 998,–; HP-Kartenleser 548,–; HP-IL-Modul 329,–; HP-Laufwerk 1435,–; Epson HX-20 1798,–; Epson RX-80 1278,–; Epson FX-80 1848,–; Sharp MZ-80B 2998,–; Sharp P5A/B 1998,–; Sharp Floppy 2698,–. Große Auswahl Papier, Farbbänder, Disketten usw. Fachhandels-Service + Fachberatung!

...auch per Post!

Wenn Sie nicht in eines unserer Ladengeschäfte kommen können, beraten und bedienen wir Sie gern per Post – schnell und bequem. Schicken Sie einfach den nebenstehenden Coupon an:

Dontenwill GmbH · Abt. CP · Steinbrückstraße 13
7880 Bad Säckingen · Telefon 07761/3095 · Telex 792339

Filiale Stuttgart:
Echterdinger Straße 9
7022 Leinfelden
Telefon 0711/750347

DONTENWILL

Filiale München:
Landwehrstraße 40
8000 München 2
Telefon 089/597993

Der heiße Computer-Coupon

an: Dontenwill · Steinbrückstraße 13 · 7880 Bad Säckingen
Schicken Sie mir bitte sofort ☐ folgende Geräte per Nachnahme
☐ folgende Geräte gegen V-Scheck anbei ☐ Gratisprospekt für:

_____

_____

Meine genaue Adresse:

_____

_____

_____ / _____

Telefon: _____ / _____

CP

RADIX HAMBURG

16 BIT-Prozessor
16 KB RAM
auf 48 KB-RAM aufrüstbar
Groß- und Kleinschreibung
direkt an jeden Fernseher anschließbar (16 Farben)
Speicherung von Daten- und Programmen auf Kassettenrecorder oder Diskettenstation
Basic als Standardsprache
Assembler, Pascal, TI-Logo möglich
umfangreiches Softwareprogramm
Anschuß an A 4-Drucker bzw. Typenradschreibmaschine möglich

Home-Computer-System TI 99/4A 599,– incl. MWSt.

Hardware

Home Computer TI 99/4 A	599,––
Peripherie-Erweiterungsbox	599,––
RS 232 Karte	399,––
Disketten Steuerkarte	499,––
Disketten Laufwerk intern	999,––
32 KB-Erweiterung	599,––
P-Code-Karte	599,––
Fernbedienung	79,––
Kassettenrecorderkabel	39,––
Kassettenrecorder spez. für Datenübertragung	149,––
Disketten, leer	9,50

Organisation und Beruf

Datenverwaltung und Analyse (M)	149,––
Statistik (M)	149,––
Text und Dateiverwaltung[1] (M)	149,––
Buchungsjournal (M)	399,––
Lagerverwaltung[1] (D)	199,––
Rechnungsstellung[1] (D)	199,––
Versandliste (D)	199,––
Elektrotechnik I (D)	115,––
Baustatik I HC (D)	115,––
Finanzberater (K)	32,––
Marketing Planspiele (K)	32,––

Spiele und Unterhaltung

Schachmeister (M)	115,––
Video-Spiele I (M)	79,––
Hallenfußball (M)	79,––
A-maze-ing, Katz und Maus (M)	79,––
Attack, Raumpatroullie (M)	79,––
Yahtzee, Kniffel (M)	79,––
Tombstone City, Wilder Westen (M)	79,––
TI-Invaders (M)	79,––
Car Wars, Autorennen (M)	115,––
Munch Man, Labyrinth (M)	115,––
Alpiner, Bergsteiger (M)	115,––
Oldies but goodies I (K)	32,––
Oldies but goodies II (K)	32,––
Adventure Modul	
Pirate Adventure (K+M)	115,––
Goldene Reise[3] (K)	39,––
Schatzsuche[3] (K)	39,––
Unmögliche Mission[3] (K)	39,––
Voodoo Castle[3] (K)	39,––
Der Graf[3] (K)	39,––
Odyssee[3] (K)	39,––
Schreckenskammer[3] (K)	39,––
Pyramide des Verhängnisses[3] (K)	39,––
Geisterstadt[3] (K)	39,––
Wilde Insel I und II[3] (K)	39,––

Aus- und Weiterbildung

Basic Lernprogramm (K)	32,––
Teach Yourself Extended Basic (K)	32,––
Basic Programm Routinen I (K)	32,––
Basic Programm Routinen II (D)	65,––
Basic Programm Routinen III (D)	65,––
Mini Memory (CMOS-RAM) (M)	299,––

Programmiersprachen

Extended Basic (M)	299,––
TI Logo[2] (M)	555,––
Editor Assembler[2] (M)	299,––
UCSD Pascal[2]	
–Compiler (D)	299,––
–Linker (D)	265,––
–Editor (D)	199,––

(M) = Modul
(D) = Diskette
(K) = Kassette

[1] nur in Verbindung mit Datenverwaltung & Analyse
[2] nur mit Peripheriebox, 32 kByte RAM-Erweiterung, Diskettens., bei PASCAL zusätzlich P-Code-Karte
[3] nur in Verbindung mit Adventure Modul

Bestell-Coupon

Absender .....

..... Tel. .....

Anz.	Bezeichnung	Preis
	Home Computer TI 99/4A	
	Kassettenkabel	

+ 5,– Versand bei unter 200,– Bestellwert _____

Summe _____

Datum _____ Unterschrift _____

Coupon an: Radix Bürotechnik, Bornstr. 4, 2000 HH 13

CC-40: der neue Basic-Computer von Texas Instruments 6 bis 18 KB-RAM Speicher, 4 Farb-Plotter anschließbar, Anwender-Software z. B. Statistik, Elektrotechnik, Mathematik, Finanzen, Physik, Sonnenenergie . . .

Radix Bürotechnik – TI Vertragshändler · Auf alle Geräte Original-Herstellergarantie. Wir führen das gesamte TI-Programm, Bitte Preisliste anfordern. · Versand per NN ab 200,– frei Haus, sonst 5,– Versandpauschale · Alle Preise incl. 13% MWSt.

Radix Bürotechnik

Bornstr. 4, 2000 Hamburg 13, Tel.: (040) 410 25 47, tägl. 9–18.30 Sa 10–13 Uhr
Filiale Kiel, Harmstr. 15, 23 Kiel, Tel.: (0431) 67 24 85, Di und Do 10–14 Uhr

Programmzugriff über das Kassettenzählwerk

Bei Programmsuchverfahren für Kassettenrecorder muß unter Umständen mit langen Ladezeiten gerechnet werden. Sinnvoller und schneller ist ein Programm, das es erlaubt, direkt über ein Bandzählwerk auf das zu ladende Programm zuzugreifen.

Das Programm, siehe Listing, dient zum gezielten Laden eines Programmes. Ein Programmsuchverfahren für Kassetten ist beim TI 99/4A zwar möglich, aber wenig sinnvoll, weil unter Umständen mit langen Ladezeiten gerechnet werden muß.

Bei diesem Programm kann über das Bandzählwerk direkt auf ein Programm zugegriffen werden.

In den Datazeilen 590 bis 670 werden die Programmnamen spezifiziert und in der Datazeile 690 werden die dazugehörigen Bandzählwerkstände eingegeben.

Das Unterprogramm von Zeile 520 bis 560 dient zum Schreiben auf dem Bildschirm und kann eventuell in anderen Programmen Verwendung finden. Die Variable Y dient als Zeilen-, X als Spaltenvariable.

Nachdem das Programm eingetippt wurde, werden in die Zeilen 580 bis 660 die Programmnamen der abzuspeichernden Programme eingetragen. Dies könnte zum Beispiel folgendermaßen aussehen:

```
570 Data 1, 2, "*** Kassetten Übersicht ***"
580 Data 4, 3, "Inhalt Nr. 1"
590 Data 6, 3, "(1) Mondlandung"
600 Data 8, 3, "(2) Startrek"
610 Data 10, 3, "(3) Tic-Tac-Toe"
```

Es ist darauf zu achten, daß alle Datazeilen mit Leerzeilen aufgefüllt werden, falls der Programmname kürzer als 23 Zeichen sein sollte. Danach wird das Kassettenübersichtsprogramm am Anfang der Kassette aufgezeichnet.

Nachdem dies geschehen ist, wird das Bandzählwerk auf 000 gestellt und das erste Programm geladen. Anschließend wird der Zählerstand notiert und das zweite Programm geladen und so weiter.

TI 99/4(A)

Ist man an das Ende gelangt (maximal 8 Programme können auf diese Art und Weise erfaßt werden), dann wird das Band zurückgespult und das Kassettenübersichtsprogramm eingelesen. Jetzt werden in der Zeile 690 die Zählwerkstände der Programme eingegeben. Die erste Zahl gibt die Programmnummer an. Die anschließenden Nullen müssen durch die ermittelten Zählwerkstände ersetzt werden (beispielsweise 690 Data 1, 000, 2, 036, 3, 057 .... etc.).

Danach kann das Kassettenübersichtsprogramm an die alte Stelle auf der Kassette abgespeichert werden.

Bild 1 gibt einen allgemeinen Überblick des Hauptprogramms (siehe Listing). Im Bild 2 sind die einzelnen Unterprogramme »Tastatur 1«, »Tastatur 2« und »Anzeige« dargestellt.

Der Ablauf zum Laden und Starten eines Pro-

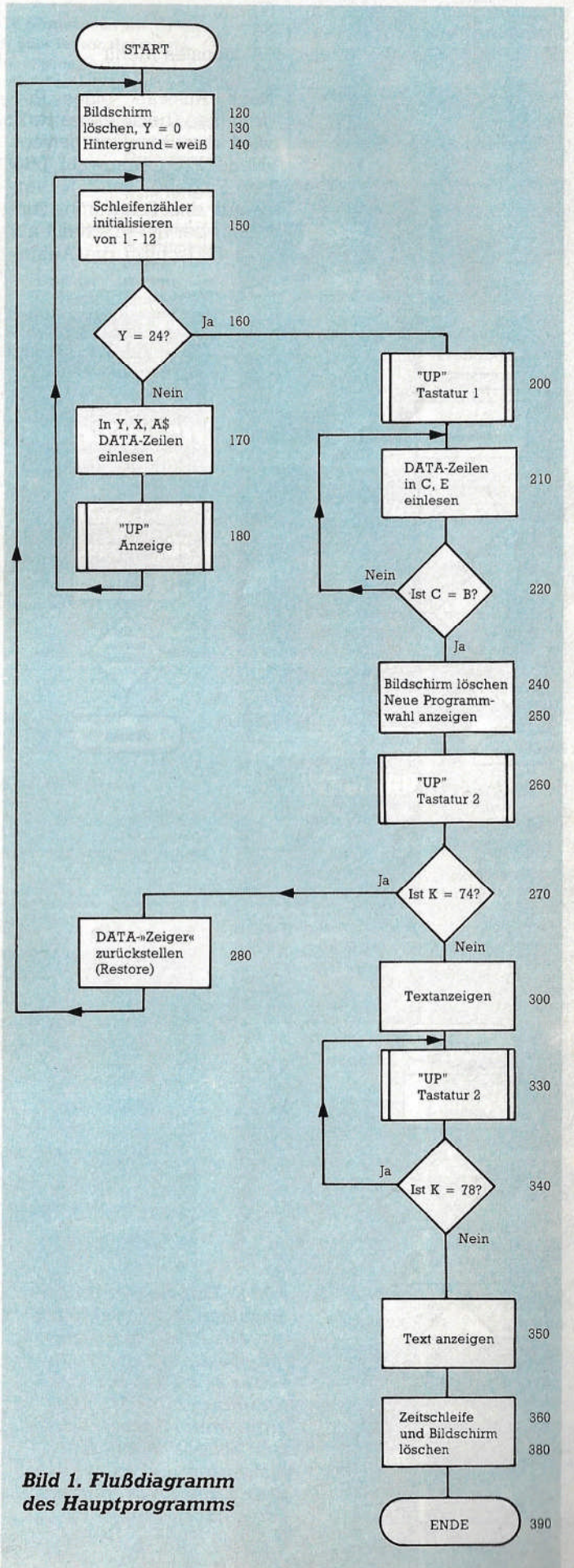


Bild 1. Flußdiagramm des Hauptprogramms

gramms sieht nun folgendermaßen aus: Zuerst erscheint die Kassettenübersicht.

Kassetten Übersicht
Inhalt: Nr. 1
(1) Mondlandung
(2) Startrek
(3) Tic-Tac-Toe
(4) .....
(5) .....
(6) .....
(7) .....
(8) .....
(9) nächstes Menü
Welches Programm?

Nach Auswahl eines Programmes (beispielsweise 2) wird der Benutzer gefragt: »Neue Programmwahl J/N« Bei Eingabe von »J« verzweigt das Programm zum Menü, bei »N« erscheint auf dem Bildschirm die Aufforderung. »Bitte stellen Sie das Zählwerk des Recorders auf 000.«
»Spulen Sie auf 36 vor.«
»Fertig (J)«.

Das heißt der Benutzer wird angewiesen, das Zählwerk auf 000 zu stellen und dann auf 36 vorzuspulen. Wenn der Benutzer damit fertig ist, tippt er »J« ein. Es erscheint auf dem Bildschirm: »Das Programm kann in wenigen Sekunden mit old CS1 geladen werden.«

Nach zirka fünf Sekunden wird der Bildschirm gelöscht und die zuvor gegebene Anweisung kann ausgeführt werden.

Damit ist das zweite Programm (Startrek) geladen.

(Gerhard Külkens)

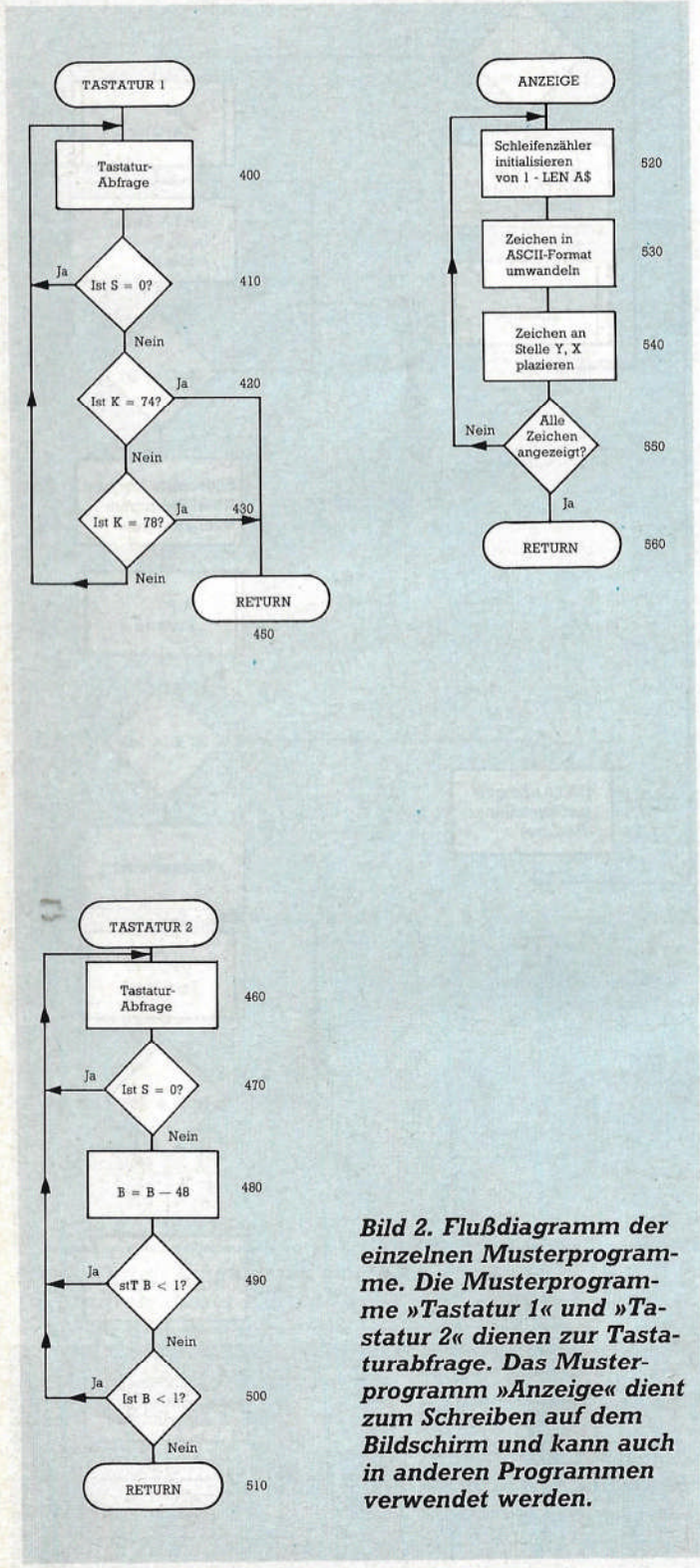


Bild 2. Flußdiagramm der einzelnen Musterprogramme. Die Musterprogramme »Tastatur 1« und »Tastatur 2« dienen zur Tastaturabfrage. Das Musterprogramm »Anzeige« dient zum Schreiben auf dem Bildschirm und kann auch in anderen Programmen verwendet werden.

```
100 REM  KASSETTENVERWALTUNG BY
110 REM  GERH. KUELKENS
         TI 99/4A TI-BASIC
         VERSION 1.1
120 CALL CLEAR
130 CALL SCREEN(16)
140 Y=0
150 FOR I=1 TO 12
160 IF Y=24 THEN 200
170 READ Y,X,A$
180 GOSUB 520
190 NEXT I
200 GOSUB 460
210 READ C,E
220 IF C=B THEN 240 ELSE 230
230 GOTO 210
240 CALL CLEAR
250 PRINT"NEUE PROGRAMMWAHL ?  J/N"::::
260 GOSUB 400
270 IF K=74 THEN 280 ELSE 300
280 RESTORE
290 GOTO 100
300 PRINT"BITTE STELLEN SIE DAS ZAEHL-":::"WERK
    DES RECORDERS AUF 000.":::
310 PRINT"SPULEN SIE AUF";E;"VOR"::
320 PRINT"FERTIG ?  (J)":::::
330 GOSUB 400
340 IF K=78 THEN 330
350 PRINT"DAS PROGRAMM KANN IN WENIGEN":::"SE-
    KUNDEN MIT OLD CS1 ":::"GELADEN WERDEN":::::
360 FOR I=1 TO 2000
370 NEXT I
380 CALL CLEAR
390 END
400 CALL KEY(0,K,S)
410 IF S=0 THEN 400
420 IF K=74 THEN 450
430 IF K=78 THEN 450
440 GOTO 400
450 RETURN
460 CALL KEY(0,B,S)
470 IF S=0 THEN 460
480 B=B-48
490 IF B<1 THEN 460
500 IF B>9 THEN 460
510 RETURN
520 FOR J=1 TO LEN(A$)
530 CODE=ASC(SEG$(A$,J,I))
540 CALL HCHAR(Y,X+J,CODE)
550 NEXT J
560 RETURN
570 DATA 1,2,"*** KASSETTEN UEBERSICHT ***"
580 DATA 4,3,"INHALT:         NR.0   "
590 DATA 6,3,"(1) ....................."
600 DATA 8,3,"(2) ....................."
610 DATA 10,3,"(3) ....................."
620 DATA 12,3,"(4) ....................."
630 DATA 14,3,"(5) ....................."
640 DATA 16,3,"(6) ....................."
650 DATA 18,3,"(7) ....................."
660 DATA 20,3,"(8) ....................."
670 DATA 22,3,"(9) NAECHSTES MENUE      "
680 DATA 24,6,"WELCHES PROGRAMM ?"
690 DATA 1,000,2,000,3,000,4,000,5,000,6,000
        7,000,8,000,9,000
```

PC 1500

Will man die Bildwirkung nicht dem Zufall überlassen, so kann man die Parameter nach einer kleinen Programmänderung auch mittels INPUT selbst eingeben und damit gestalterisch auf das Bild Einfluß nehmen. Zeile 650 sorgt dafür, daß nichts »aus dem Rahmen fällt«, was bei den Möwen allerdings auch sehr eigenwillig aussehen kann (Y1 in Zeile 540 vergrößern). Das Listing wurde ebenfalls mit dem Drucker CE 150 mit Hilfe eines ROTATE-LIST-Programms streifenweise (10 Zeilen je Streifen) erstellt.

(Manfred Schmidt)

```
 10 REM       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 15 REM       @                               @
 20 REM       @     " R E G A T T A "         @
 30 REM       @    M. SCHMIDT, HAMBURG        @
 40 REM       @        MAERZ  1983            @
 45 REM       @                               @
 50 REM       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 60 REM
100 "A":RANDOM :WAIT 0
110 CLS :PRINT "BILDERZAHL (SERIE): ";:INPUT N
120 N=INT N:IF N<1 GOTO "A"
130 CLS :PRINT "  L E I N E N   L O S !"
140 FOR BILD=1 TO N
150 S=RND 6+3:REM Anzahl der Schiffe im Bild = 4...9
160 B=RND 3:REM Segelbreite = 1...3
170 U=RND 0-0.5:REM Segel-Unterkantenschraege = -0.5...+0.5
180 M=RND 5+5:REM Anzahl der Moeven im Bild = 6...10
190 W=RND 15-8:REM Windstaerke und -richtung = -7...+7
200 REM
201 REM       R A H M E N :
202 REM
210 GRAPH :GLCURSOR (0,-165):SORGN
220 LINE (0,-55)-(215,160),0,0,B
300 REM
301 REM       BESCHRIFTUNGEN:
302 REM
310 GLCURSOR (0,-80):LPRINT "REGATTA ";BI;"/";STR$ N
320 GLCURSOR (120,-50):CSIZE 1:LPRINT "Name          JAHR"
400 REM
401 REM       H O R I Z O N T :
402 REM
410 LINE (215,0)-(0,0),3
500 REM
501 REM       M O E V E N :
502 REM
510 FOR I=1 TO M
520 GR=1.5*(1+RND 0):REM unterschiedliche Groesse der Moeven
530 X1=10+RND 180:X2=X1+2*GR:X3=X2+GR:X4=X3+GR:X5=X4+2*GR
540 Y1=75+RND 75:Y2=Y1+GR/2:Y3=Y2-GR:Y4=Y2:Y5=Y1
550 LINE (X1,Y1)-(X2,Y2)-(X3,Y3)-(X4,Y4)-(X5,Y5),0,0
560 NEXT I
600 REM
601 REM       S C H I F F E :
602 REM
610 FOR I=1 TO S
620 X1=10+RND 190:Y1=1+RND 40
630 X2=X1+Y1*B:Y2=Y1*(1+U)
640 X3=X1+Y1*(B+W)/2:Y3=Y1*3
650 IF X2>205 OR Y2>42 OR X3>205 OR X3<10 LET I=I-1: GOTO "next"
660 LINE (X1,-Y1)-(X2,-Y2)-(X3,Y3)-(X1,-Y1),0,1+2.99*I/S
670 "next":NEXT I
700 REM
710 GLCURSOR (0,-180):NEXT BILD
720 TEXT :COLOR 0:BEEP 3:CLS
730 WAIT :PRINT " E N D E ":END
```

Listing des Programms »Regatta«

Bibliotheksverwaltung noch schneller

TI 99/4

Das in CP 4/83 auf Seite 37 ff. abgedruckte Programm »Bibliotheksverwaltung« mit dem TI 99/4 habe ich getestet und dabei festgestellt, daß die Module »Einlesen« und »Speichern« bei der Verwendung eines Kassettenrecorders sehr zeitaufwendig sind; pro erfaßtes Buch sind zirka 80 Sekunden erforderlich. Nun sind Kassettenrecorder zwar immer etwas langsam, doch durch folgende Änderungen in Zeile 8060 (auf Seite 38) und Zeile 10040 (auf Seite 40) läßt sich die Zeit jeweils auf etwa ⅙ (also zirka 80 Prozent weniger) verringern:

Zeile: 8060: PRINT # 2 : SF$(I,1),
SF$(I,2), SF$(I,3), SF$(I,4), SF$(1,5), SF$(1,6), SF$(I,7), SF$(I,8), SF$(I,9), SF$(I,10)

Zeile: 10040 INPUT # 3 : SF$(I,1), SF$(I,2), SF$(I,3), SF$(I,4), SF$(I,5), SF$(I,6), SF$(1,7), SF$(I,8), SF$(I,9), SF$(I,10)

Die Zeitersparnis entsteht dadurch, daß jeder PRINT-Befehl für den Speicher erst eine Kennung erforderlich macht. Beim Durchlaufen einer FOR-NEXT-Schleife wird aber für jede der 10 Zeilen eines Buchtitels erneut der PRINT-Befehl aktiviert, während durch die Änderung zwar das Programm etwas aufwendiger wird, was aber durch die Zeitersparnis mehr als wettgemacht wird.

(Reinhard Sundermann)

TI CC 40 — kleiner Computer ganz groß

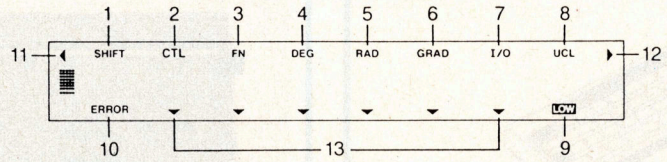
Eigentlich bin ich ja kein Freund von Taschencomputern. Denn einige dieser Kleinstcomputer rücken doch schon bedenklich in die Nähe von Taschenrechnern. Deshalb war ich zunächst gar nicht begeistert, als es hieß, der TI Compact Computer 40 müsse getestet werden. Doch wie so häufig, täuschte der erste Eindruck: Der CC 40 ist leistungsfähiger, als seine Größe vermuten läßt.

Bis Ende letzten Jahres war Texas Instruments im Heimcomputerbereich nicht besonders erfolgreich. Doch mit einem neuen Konzept und einem überarbeiteten TI 99/4A schaffte TI dann auch in Deutschland den Durchbruch. Insgesamt wurden im letzten Jahr in Deutschland ungefähr 100000 Heimcomputer verkauft, für dieses Jahr wird mit einer Verdreifachung gerechnet. Texas Instruments will seine Stückzahlen sogar verfünffachen, und es mit neuen Produkten, die das Sortiment abrunden sollen, gar auf das Achtfache bringen. Zu diesen neuen Produkten gehören der TI 99/2, der als Konkurrenz zum Sinclair ZX81 gedacht ist, und der TI Compact Computer 40, der im folgenden ausführlicher vorgestellt wird.

Zunächst einmal die technischen Daten des CC 40. Er ist 240 x 145 x 25 mm groß, oder besser klein, und wiegt ungefähr 600 Gramm. Er verfügt über eine einzeilige LCD-Anzeige mit 31 Zeichen und 18 Indikatoren, von denen sechs von Programmen aus gesteuert werden können (Bild 1). Die Zeichen werden in einer 5 x 8 Matrix dargestellt. Der Kontrast der Anzeige läßt sich über einen Regler an der linken Seite einstellen (Bild 2). Die Tastatur ist eine verkleinerte QWERTY-Tastatur mit separatem Zahlenfeld (Bild 3). Jede Taste hat eine automatische Wiederhofunktion, wenn sie etwas länger gedrückt wird. Der Zeichensatz umfaßt 189 Zeichen, sechs Zeichen können über den Basic-Befehl CHAR umdefiniert werden. Ein Beispiel dafür ist in Listing 1 zu sehen. Hier werden einige Zeichen so umdefiniert, daß sie zusammen das Bild eines Zuges ergeben.

Der CC 40 arbeitet mit dem CMOS-Mikroprozessor TMS 70C20, einem 8-Bit-Prozessor mit internen 2 KByte ROM und 128 Byte RAM. Betriebssystem und Basic-Interpreter belegen zusammen 34 KByte ROM. Der Benutzerspeicher ist in der Grundausstattung 6 KByte groß, und mit Steckmodulen erweiterbar auf 18 KByte. Außer Speicher können diese Modulkassetten auch Software in ROMs enthalten, deren maximaler Umfang 128 KByte betragen kann. Der Prozessor kann mit Bank-Switching auf 168 KByte maximal zugreifen; eine erstaunliche Zahl für ein so kleines Gerät. Es sind derzeit Softwaremodule für die Bereiche Mathematik, Statistik, Finanzwesen und Ingenieurwissenschaften erhältlich (Bild 4).

Die Stromversorgung erfolgt normalerweise über vier Batterien, für die eine Lebensdauer von zirka 200 Stunden angegeben wird; es ist aber auch ein Anschluß für einen Netzadapter vorhanden. Der CC 40 schaltet sich automatisch aus, wenn er länger als zehn Minuten nicht benutzt wird. Es gehen dabei jedoch keine Programme verloren, da der Speicher auch weiterhin mit Strom versorgt wird. Nach erneutem Ein-



Indikator	Bedeutung
1. SHIFT	Shift-Taste wurde gedrückt
2. CTL	Control-Taste wurde gedrückt
3. FN	Funktionstaste wurde gedrückt
4. DEG	Anzeige in Grad, Winkel oder Bogenmaß
5. RAD	
6. GRAD	
7. I/O	Ein-/Ausgabe läuft
8. UCL	Großschreibung (Upper Case Lock)
9. LOW	Batterie muß gewechselt werden
10. ERROR	Ein Fehler ist aufgetreten
11. ◄	Es befindet sich weiterer Text auf der linken Seite
12. ►	Es befindet sich weiterer Text auf der rechten Seite
13. ▼	Programmierbare Indikatoren

Bild 1. Die 31-Zeichen-Anzeige des CC 40 mit verschiedenen Indikatoren.

schalten ist ein Programm also sofort wieder verfügbar und braucht nicht erst wieder geladen werden. Auf der Rückseite läßt sich ein kleiner Ständer herausklappen, um den Rechner besser positionieren zu können.

Peripherie wird über den Texas HEX-Bus angeschlossen. Zur Zeit gibt es einen Vierfarb-Drukker/Plotter, ein Wafertape-Magnetband-Laufwerk (beide auf Bild 5) und eine RS232C/Parallel-Schnittstelle. Die RS232C-Schnittstelle ermöglicht den Anschluß an größere Systeme und Drucker. Es sind weitere Geräte für den HEX-Bus geplant, zum Beispiel ein TV-Interface, mit dem dieser Taschencomputer an einen Schwarzweiß-Fernseher angeschlossen werden kann. Alle Peripheriegeräte arbeiten wie der Rechner batteriebetrieben. Mit Hilfe eines Adapters läßt sich diese Peripherie übrigens auch an den TI 99/4A anschließen. Geräte mit HEX-Bus-Schnittstelle werden miteinander mit einem achtpoligen Kabel verbunden. Der Computer hat einen Anschluß, jedes Peripheriegerät zwei, so daß die Geräte miteinander verkettet werden können. Das erste Peripheriegerät wird in den Computer eingesteckt, das zweite in das erste und so weiter.

Der Printer/Plotter HX-1000 druckt mit vier kleinen Kugelschreibern in den Farben schwarz, blau, grün und rot mit einer Geschwindigkeit von 12 Zeichen pro Sekunde. Geschrieben wird auf einfache Kassenrollen mit 58 mm Breite. Es können verschiedene Schriftgrößen produziert werden, wobei in eine Zeile bis zu 36 Zeichen passen. Farbgrafiken (Bilder 6 und 7) lassen sich über einfache Befehle erzeugen.

Das Wafertape HX-2000 ist als preiswerte Alternative zu Disketten-Laufwerken und als schnellere Variante einfacher Kassettenrecorder gedacht. Ein Wafer ist eine Mikrokassette mit schmalem Endlosband (Bild 8). Das Band gibt es in verschiedenen Längen, so daß die Speicherkapazität zwischen 6 und 48 KByte liegen kann. Als maximale Zugriffszeit werden 1½ Minuten angegeben. Auf Daten kann nur sequentiell zugegriffen werden. Alle Funktionen des Bandlaufwerks vom Formatieren bis zum Laden von Programmen werden vom CC 40 unterstützt.

Wer weitere Peripherie anschließen möchte, zum Beispiel einen Matrixdrucker, oder Daten zu größeren Rechnern übertragen möchte, kann dies mit Hilfe der RS232C-/Parallel-Schnittstelle HX-3000 tun. Sie verfügt über Baudraten im Bereich von 50 bis 19200, wählbare Anzahl von Daten- und Stopbits und verschiedene Parameter. Die Schnittstelle ist leistungsfähiger als bei vielen Tischcomputern und wird dadurch gesteuert, daß beim OPEN-Befehl die Parameter mit angegeben werden.

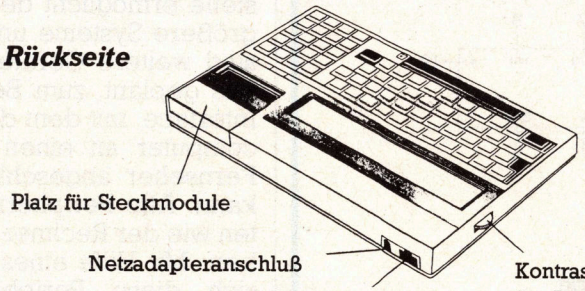


Bild 2. Die Rückseite des CC 40.

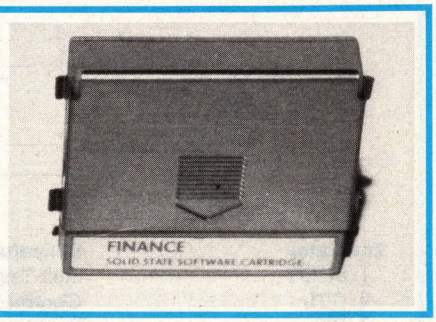


Bild 4. Ein Softwaremodul

```
100 ! ZUG PROGRAMM
110 CALL CHAR(0,"0
10001001D151F0A")
120 CALL CHAR(1,"0
00000001F151F0A")
130 CALL CHAR(2,"0
000000011151F0A")
140 CALL CHAR(3,"0
0000000000111F0A")
150 PRINT "STOP MI
T BELIEBIGER TASTE
":PAUSE 1.5
160 FOR A=1 TO 31
170 DISPLAY AT(A)B
EEP,CHR$(3)&CHR$(2
)&CHR$(1)&CHR$(0)
180 CALL KEY(K,S):
IF S=0 THEN NEXT A
:GOTO 160:END
```

Listing 1. Dieses »Zug-Programm« zeigt die Verwendung von selbsterstellten Zeichen

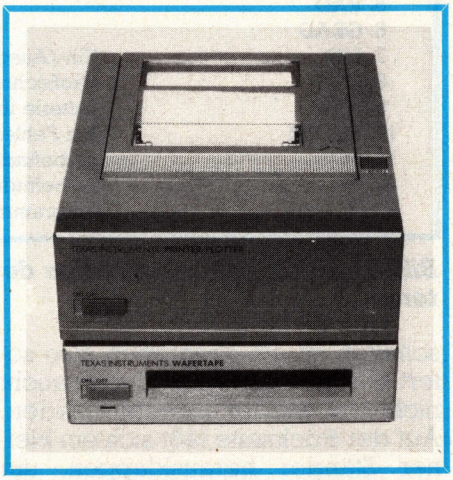


Bild 5. Printer/Plotter und Wafertape-Laufwerk

Die Preise liegen in erschwinglichen Bereichen. Der CC 40 selbst kostet 748 Mark, das Wafertape 448 Mark, der Printer/Plotter 548 Mark und die RS232C-Schnittstelle 498 Mark, alle Preise inklusive Mehrwertsteuer.

Der Compact Computer 40 ist in erweitertem Basic und in Assembler programmierbar, zusätzlich aber auch als normaler Taschenrechner mit allen technisch-wissenschaftlichen Funktionen verwendbar. Und eben für technisch-wissenschaftliche Anwendungen und auf längere Sicht als Ersatz für den magnetkarten-programmierbaren Rechner TI 59 ist der CC 40 gedacht.

Das Basic entspricht bis auf die Befehle für Farbe, Grafik und Musik in etwa dem Extended Basic des TI 99/4A, das besonders wegen seiner umfangreichen Befehle und Möglichkeiten bekannt ist. Einige Befehle wurden sogar noch etwas erweitert; so zum Beispiel der ACCEPT-Befehl. ACCEPT ist eine Texas-Instruments-Variante des altbekannten INPUT (das natürlich auch noch vorhanden ist). Bei ACCEPT können verschiedene Parameter angegeben werden: die Spalte, in der die Eingabe erfolgen soll (AT), der Wertebereich der Eingabe (SIZE), eine Überprüfung auf numerisch, alphanumerisch, nur Großbuchstaben und Kombinationen davon oder auch beliebige selbst definierbare Überprüfungen, zum Beispiel kann nur J und N zugelassen werden (VALIDATE), ein Standardwert, der genommen wird, wenn nichts eingegeben wird (NULL), Löschen des Bildschirms vor der Eingabe (ERASE ALL) und ein Piepston als Eingabeaufforderung (BEEP). Wie komfortabel der

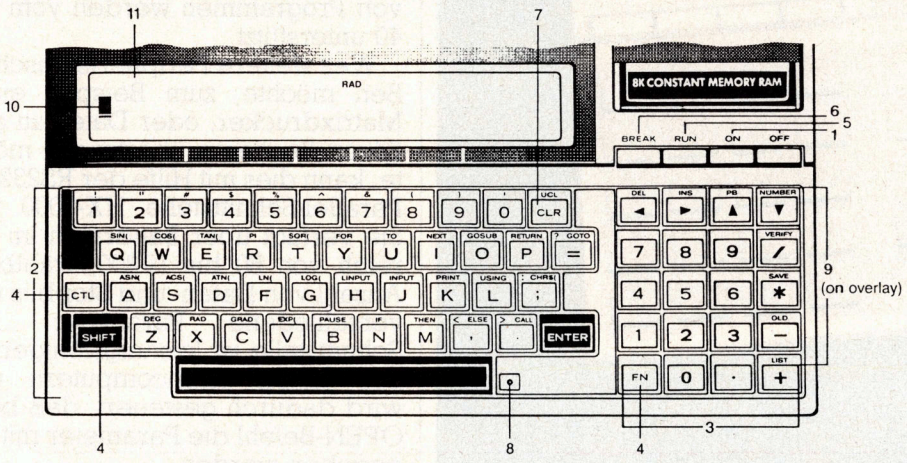


Bild 3. Die Tastatur

1. Ein- und Ausschalter
2. Schreibmaschinenteil
3. separates Zahlenfeld
4. Shift-, Funktions- und Control-Taste
5. RUN-Taste
6. BREAK-Taste
7. Clear-Taste (Anzeige löschen)
8. Reset-Knopf (versenkt)
9. Basic-Worte (auf Folie)
10. Cursor
11. Anzeige

Befehlssatz ist, ließe sich noch an vielen Beispielen demonstrieren. Für den Benutzer bedeutet dies zum einen, daß er es bei der Programmierung wesentlich leichter hat, da er solche Befehle wie RESTORE N nicht umständlich selbst in Assembler programmieren muß, und zum anderen, daß er aus den 6-KByte-Speichern wesentlich mehr herausholen kann als auf anderen Rechnern mit weniger komfortablen Basic-Versionen.

Fehlermeldungen und Systemanfragen können in verschiedenen Sprachen ausgegeben werden. Dies wird mit Hilfe der Basic-Funktion SETLANG bewerkstelligt. CALL SETLANG (1) schaltet zum Beispiel auf deutschen Dialog um. Statt eines »Variable not defined« erscheint dann also das verständlichere »Variable nicht definiert«. Diese Umschaltung erstreckt sich auch auf die Softwaremodule, deren Abfragen und Meldungen in deutsch und englisch programmiert sind und die sich mit diesem Befehl ebenfalls auf die gewünschte Sprache einstellen lassen.

Die interne Rechengenauigkeit beträgt 13 Stellen, in der Anzeige werden davon jedoch nur zehn Stellen verwendet. Variablennamen können bis zu 15 Zeichen lang sein. Wenn mehreren Variablen der gleiche Wert zugewiesen werden soll, dann kann das zum Beispiel so geschehen: X,Y,Z=132. Nach dieser Zuweisung haben alle drei Variablen den Wert 132. Auch bei der Eingabe durch INPUT beziehungsweise ACCEPT können alle Taschenrechnerfunktionen verwendet werden. Wird zum Beispiel nach dem Jahresgehalt gefragt, dann muß man das nicht erst im Kopf ausrechnen, sondern man kann gleich 13 x 1695,30 eingeben. Das funktioniert nicht nur mit den vier Grundrechnungsarten, sondern auch mit trigonometrischen und Wurzelfunktionen, man kann also ohne weiteres eingeben: PI*10.45^2; in der entsprechenden Input-Variablen steht hinterher der richtige Wert.

Der Aufruf von Unterprogrammen erfolgt über Namen, wobei jede Routine ihre eigenen lokalen Variablen verwendet; es können also in einem Unterprogramm die gleichen Variablennamen verwendet werden, wie im Hauptprogramm, ohne daß sie sich gegenseitig beeinflussen.

Das Basic ist meines Erachtens besonders ideal für Anfänger zum Erlernen der Sprache, da es sehr komfortabel und übersichtlich ist und nicht ständig verlangt, daß man sich für irgendeine simple und selbstverständliche Funktion selbst den Kopf zerbrechen muß. Der Benutzer kann sich voll und ganz auf sein Programm konzentrieren und wird nicht durch die Entwicklung von Tricks und Kniffen davon abgelenkt.

Die Editierfunktionen stehen dem hohen Niveau der Basic-Sprache in nichts nach. Mit Hilfe der FN-Taste lassen sich die wichtigsten Basic-Befehle durch zwei Tastendrucke generieren. Zehn Tasten können mit eigenen Texten belegt werden, und zwar 0 bis 9, die dann in Verbindung mit der FN-Taste wieder abgerufen werden können. Mit den Cursortasten kann man sich beliebig in einem Programm bewegen, wobei die 31-Zeichen-Anzeige wie ein Fenster auf einen Bildschirm zu verstehen ist, der so groß ist, wie das Programm selbst. Mit NUM kann man sich die Zeilennummern vorgeben lassen (in anderen Basics AUTO). Mit DELETE können Zeilenbereiche gelöscht werden. Auch der Befehl RENUMBER zur Umnumerierung des Programms ist vorhanden. Bei der Arbeit mit der Tastatur ist nur eines zu Beginn störend: man muß sich erst daran gewöhnen, daß die Eingabe-Taste dort liegt, wo gewöhnlich die rechte Shift-Taste untergebracht ist. Dafür ist aber die linke Shift-Taste so ausgelegt, daß der Shift-Status nach dem Drücken eingeschaltet bleibt, bis eine andere Taste betätigt wurde. Zeichen können so auch nur mit einer Hand eingegeben werden. Das gleiche gilt für die Control- und die FN (Funktions)-Taste.

Der Texas-Instruments-Compact-Computer 40 ist ein vielseitiger und leistungsfähiger Kleincomputer, den man sowohl als kleinen Tischcomputer (mit der Peripherie) oder als Taschencomputer für Schule, Uni, Büro und Zuhause verwenden kann. Er hat mich davon überzeugt, daß auch in einem so kleinen Gehäuse echte Computerleistung untergebracht sein kann. Ich kann ihn mit ruhigem Gewissen jedem empfehlen, der Wert auf möglichst viel Leistung auf kleinstem Raum legt oder sich mit der Programmierung in Basic vertraut machen möchte. Der CC 40 ist nur äußerlich ein »kleiner« Computer.

(jü)

Bilder 6. und 7. Farbgrafiken des Printer/Plotter

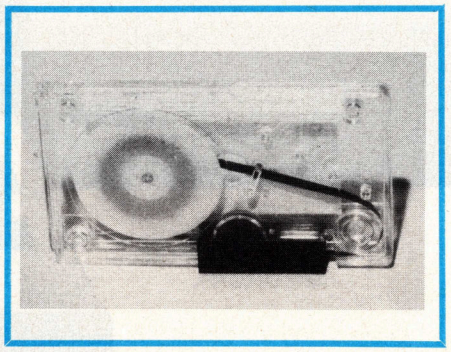


Bild 8. Ein Wafertape (Endlosband)

Fakturierung selbst entwickelt

So kann jeder seine eigene Fakturierung auf seinem Computer selbst schreiben — auch der Anfänger. Der Artikel bezieht sich auf ein Minimal-Basic, welches in allen Computern identisch ist. Stellen, wo computerspezifische Eigenschaften genutzt werden können, werden im Text gesondert erwähnt.

Eine Fakturierung läuft in der Regel wie in Bild 1 dargestellt. Ob alle erwähnten Teilbereiche einer Rechnung für jeden zutreffen oder bestimmte Ergänzungen, die hier nicht aufgeführt sind, noch durchzuführen sind, ist vom Einzelfall abhängig. Vorausgesetzt für unsere Fakturierung werden die in einer Datei abgelegten Kundenstammdaten und Artikeldaten. Außerdem sollen alle offenen Posten auch in einer Datei zusammengefaßt sein.

Um auch rechner- und druckerspezifische Eigenschaften nutzen zu können, wie etwa Breitschrift bei Druckern, wollen wir dem Programm eine Definition von Steuerzeichen voranstellen, wie sie in Bild 2 gegeben ist. Der Code, mit dem diese Steuerzeichen zu versehen sind, ist den jeweiligen Handbüchern zu entnehmen.

Beginnen wir jetzt jedoch mit der Programmentwicklung. Zunächst werden ein paar allgemeingültige Variablen benötigt, wie zum Beispiel der Unterstreichungsstrich, den wir wie folgt definieren wollen:

```
1000 SU$ = "—"
1010 FOR I = 1 TO 7
1020    SU$ = SU$ + SU$
1030 NEXT
1040 SU$ = LEFT$ (SU$,80)
```

Des weiteren benötigen wir einen Zeilenzähler (ZZ), in dem wir festhalten, wieviele Zeilen bis zum unteren Papierrand noch vorhanden sind. Bei Druckern mit sogenanntem Formfeed ist dieser Zeilenzähler nicht notwendig, aber nicht jeder Drucker ist damit ausgerüstet. Die übliche Seitenlänge für eine Rechnung beträgt 72 Zeilen. Des weiteren brauchen wir ein Feld von Zeichenreihen, in dem wir die zu druckenden Daten während des Erstellens der Rechnung zwischenspeichern wollen, um nachher nur noch dieses Textfeld drucken zu müssen. Möglich wäre es auch, jede einzelne erarbeitete Zeile vom Rechner sofort drucken zu lassen, jedoch ist dann mit den Daten immer nur ein durchlaufen möglich. Bei Verwendung eines Thermodruckers ohne Durchschläge ist man manchmal jedoch darauf angewiesen, die ermittelten Daten zweimal zu drucken. Daher die Zwischenspeicherung in der Variablen TE$(72).

```
1050 ZZ = 72
1060 DIM TE$(72)
```

Als nächstes wollen wir davon ausgehen, daß das gültige Tagesdatum und die letzte Rechnungsnummer in zwei getrennten sequentiellen Dateien gespeichert sind, da eine Fakturierung als Programm selten alleine steht.

Nummern sollten manuell zu ändern sein

Bei der Rechnungsnummer sollte man jedoch noch eine manuelle Änderung durch den Bediener zulassen, da eventuell auch Rechnungen mit der Hand geschrieben werden und somit keine fortlaufende Rechnungsnummer gewährleistet ist.

```
1070 Öffne sequentielle Datei 'Datum' auf Kanal 1
1080 INPUT# 1,DA$
1090 Schließe Datei 'Datum'
1100 Öffne sequentielle Datei 'Rechnungsnummer'
1110 INPUT# 1,RN
1120 Schließe Datei 'Rechnungsnummer'
1130 RN = RN + 1
1140 PRINT"Aktuelle Rechnungsnummer (";rn;") ändern ?"
1150 PRINT" (j/n)"
1160 GETA$
1170 IF A$ = "" THEN 1160
1180 IF A$ = "j" THEN INPUT "Neue Nummer";RN : GOTO 1210
1190 IF A$ = "n" GOTO 1210
1200 GOTO 1160
```

An dieser Stelle könnte man die neue Rechnungsnummer schon wieder in der Datei 'Rechnungsnummer' abspeichern, jedoch wird bei einem Rechnerabsturz im weiteren Verlauf diese Nummer nicht mehr herangezogen wegen 'RN=RN+1' in Zeile 1130. Deshalb ist es ratsam, die Rechnungsnummer erst nach korrekter Durchführung der gesamten Rechnung abzuspeichern.

Als nächstes muß die Kundennummer eingelesen und überprüft werden, ob zu der eingegebenen Nummer überhaupt ein Datensatz exisitiert. Dazu muß man auf die speziellen Belange der verwendeten Dateiverwaltung eingehen und die Daten genauso suchen, wie es in der Dateiverwaltung zum Beispiel zum Ändern der Daten geschieht. In einfachen Dateiverwaltungen ist die Kundennummer zumeist auch die physikalische Nummer des Datensatzes. Dies ist der einfachste und schnellste Zugriff, hat aber den Nachteil, daß die Kundennummern von 1 beginnend fortlaufend vergeben werden müssen, und keine Alpha-Zeichen zulassen. Das Programmstück sieht dann wie folgt aus:

```
1210 Öffne Datei 'Kunden' auf Kanal 1
1220 INPUT" Kundennummer ";KN
1230 Setze Zeiger auf Datensatz mit Nummer KN zum Beispiel mit dem Befehl RECORD# 1, (KN)
1240 Lies Datensatz entweder als ganzes mit INPUT# 1,R$ oder, wenn er mit Trennzeichen versehen ist, INPUT# 1,NA$,VN$,PL$, O$,KO,UM ...
```

Im weiteren sind eventuelle Fehlermeldungen von den Floppys abzuprüfen (zum Beispiel RECORD NOT PRESENT) oder logische Löschvorgänge zu erkennen. Bei einem Löschvorgang in der Dateiverwaltung werden meist gelöschte Sätze nur als solche gekennzeichnet (mit einem Sonderzeichen an der ersten Stelle des Namens). Die Prüfungen könnten also aussehen:

```
1250 IF 'Floppy-Fehler' THEN 1220
1260 IF LEFT$(NA$,1) = "@" THEN 1220
1270 Schließe Datei 'Kunden'
```

Bei Erkennen eines Fehlers kann auch noch eine Fehlermeldung

ausgegeben werden, etwa mit
1250 IF 'Floppy-Fehler' THEN 10000
1260 IF LEFT$(NA$,1) = "@" THEN 10000
und
10000 PRINT:PRINT:PRINT:PRINT CHR$(7) :REM BELL
10010 PRINT" Kunde nicht vorhanden "
10020 GOTO 1220

Angenommen wurde hier, daß ein logisch gelöschter Kunde an erster Stelle des Namen ein '@' enthält.

Kann der Anwender seine Kundennummer frei wählen, so muß im einfachsten Fall auch die Direktzugriffsdatei sequentiell durchsucht werden mit:
(AZ = Anzahl der Kunden
KN$ = abgespeicherte Kundennummer
EK$ = eingelesene Kundennummer)
1210 Öffne Datei 'Kunden' auf Kanal 1
1220 INPUT" Kundennummer ";EK$
1230 FOR I = 1 TO AZ
1240 Setze Zeiger auf Datensatz mit Nummer I zum Beispiel mit dem Befehl RECORD# 1,(I)
1250 Lies Datensatz entweder als Ganzes mit INPUT# 1,R$ oder, wenn er mit Trennzeichen versehen ist, INPUT# 1,NA$,VN$,S$,PL$,O$,KO,UM.....
1260 IF KN$ = EK$ THEN 1320
1270 NEXT
1280 PRINT:PRINT:PRINT:PRINT CHR$(7) :REM BELL
1290 PRINT" Kunde nicht vorhanden "
1300 GOTO 1220
1310 Schließe Datei 'Kunden'

Bei Verwendung von anderen Suchverfahren (Binär-Bäume, B-Bäume, AVL-Bäume, Hashing) sind am besten die jeweiligen Programmstücke zum Auffinden eines Datensatzes zu kopieren, da die Programmierung meist aufwendiger ist.

Ist der Kunde gefunden, so sollten dessen Daten am Bildschirm angezeigt werden, und der Briefkopf kann gedruckt werden.

Anzeigen am Bildschirm:
1400 PRINT" Kundennummer : ";KN$
1410 PRINT" Name: ";NA$
1420 PRINT" Vorname: ";VN$
1430 PRINT" Straße: ";S$
1440 PRINT" PLZ/Ort: ";PL$;" ";O$
1450 PRINT" Kontostand: ";KO
.
.

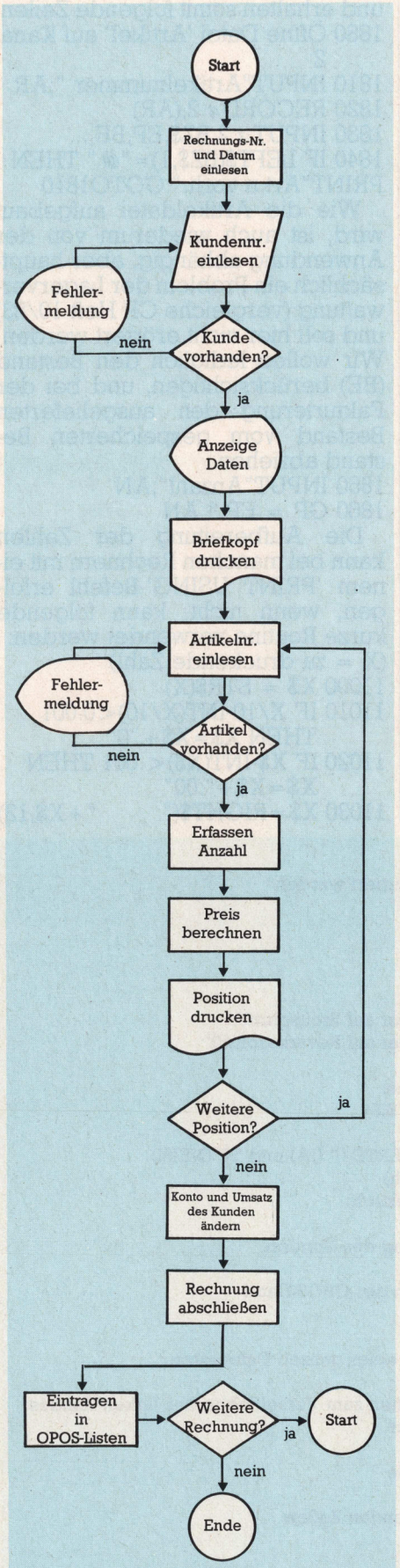


Bild 1. Flußdiagramm für eine allgemeine Fakturierung

.
Briefkopf:
1500 OPEN4,4 :REM Drucker als Datei öffnen
1510 FOR I = 1 TO 12 :REM Leerzeilen bis zum Brieffenster
1520 PRINT# 4
1530 NEXT
1540 RA$=" "
1550 PRINT# 1,RA$;"Herrn/Frau/Frl./Firma"
1560 PRINT# 1,RA$;VN$;" , ";NA$
1570 PRINT# 1,RA$;S$
1580 PRINT# 1
1590 PRINT# 1,RA$;PL$;" ";O$
1600 FOR I = 1 TO 5
1610 PRINT# 1
1620 NEXT
1630 PRINT# 1,RA$;DB$;"Rechnung"
1640 PRINT# 1,RA$;DB$;" ----------"
1650 PRINT# 1,SPC(55);"Rechnungsnr.:";RN
1660 PRINT# 1,SPC(55);"Kundennr.:";KN$
1670 PRINT# 1,SPC(55);"Datum";DA$
1680 PRINT# 1,SPC(55);DR$;"Bei Zahlung bitte angeben"
1690 PRINT# 1:PRINT# 1

Dabei werden in den Zeilen 1550 bis 1620 die Adresse und ein paar abgrenzende Leerzeilen gedruckt. Die Variable RA$ ermöglicht es, den Abstand zum linken Rand einfach zu bestimmen, ohne daß beim Testen jeweils alle Zeilen, die einen Ausdruck beinhalten, geändert werden müssen.

Der Rest druckt den eigentlichen Kopf der Rechnung, wobei besonders auf die Verwendung von DB$ und DR$ zu achten ist. Zum allgemeinen Teil einer Rechnung gehört auch noch die Titelzeile, die wiederum den Bedürfnissen der einzelnen Anwendung angepaßt sein muß. Wir wollen uns hier auf die Daten, Anzahl, Bezeichnung, Einzelpreis und Gesamtpreis beschränken:
1700 PRINT# 1,RA$"Anzahl Bezeichnung ";
1710 PRINT# 1,"E-Preis G-Preis"
1720 PRINT# 1,RA$;LEFT$(SU$,60)
1730 PRINT# 1,
1740 ZZ = ZZ — 34
1750 SU = 0

Nachdem die Zahl der noch verbleibenden Zeilen auf der Seite an ZZ übergeben wurden (bis hierher wurden bereits 34 verbraucht) und die Variable für die Rechnungssumme mit '0' vorbesetzt wurde, kann zur Erfassung der einzelnen Rechnungspositionen übergegangen werden. Auch hier gibt es wie-

der mehrere Möglichkeiten, die vom Anwendungsfall abhängig sind. Als erste Entscheidung muß man die Herkunft der Artikeldaten ansehen: Handeingabe oder Verwendung einer Artikeldatei. Zuerst wollen wir uns der manuellen Eingabe zuwenden, da diese am einfachsten zu realisieren ist:

```
1800 INPUT"Anzahl ";AN
1810 INPUT"Bezeichnung ";BE$
1820 INPUT"Einzelpreis ";EP
1830 GP = AN * EP'
1840 GOSUB 'Drucken' (s.u.)
1850 INPUT"Weitere Position";JN$
1860 IF JN$="J" OR JN$ = "j" THEN
1800
1870 IF JN$="N" OR JN$ = "n"
THEN 'Nachbehandlung'(s.u.)
1880 GOTO 1850
```

Rechner sind jedoch auch dazu da, Arbeitsabläufe zu rationalisieren, und deshalb ist es bei festem Artikelstamm günstiger, auch diese in einer Datei (Direktzugriff) abzulegen und nur über eine Artikelnummer aufzurufen. Die Problematik des Auffindens wurde schon bei den Kunden besprochen und verläuft bei den Artikeln genauso. Der Einfachheit halber wollen wir annehmen, daß die Artikelnummer der Datensatznummer entspricht, und erhalten somit folgende Zeilen:

```
1880 Öffne Datei 'Artikel' auf Kanal
     2
1810 INPUT"Artikelnummer ";AR
1820 RECORD# 2,(AR)
1830 INPUT# 2,BZ$,EP,BE,...
1840 IF LEFT$(BZ$,1)="@ " THEN
PRINT"Art.n.vorh.":GOTO1810
```

Wie die Artikeldatei aufgebaut wird, ist auch wiederum von der Anwendung abhängig, aber hauptsächlich ein Problem der Lagerverwaltung (vergleiche CP Heft 10/83) und soll hier nicht erörtert werden. Wir wollen lediglich den Bestand (BE) berücksichtigen, und bei der Fakturierung den ausgelieferten Bestand vom gespeicherten Bestand abziehen.

```
1850 INPUT"Anzahl";AN
1860 GP = EP * AN
```

Die Aufbereitung der Zahlen kann bei manchen Rechnern mit einem 'PRINT USING'-Befehl erfolgen, wenn nicht, kann folgende kurze Routine verwendet werden:

```
(X = zu druckende Zahl)
11000 X$ = STR$(X)
11010 IF X/10-INT(X/10)<0.001
      THEN X$=X$+"0"
11020 IF X$-INT(X$)<.001 THEN
      X$=X$+".00"
11030 X$=RIGHT$("          "+X$,12)
11040 X$=LEFT$(X$,9)+","
      +RIGHT$ (X$,2)
11050 IFX>999.99 THEN X$=
      LEFT$(X$,6)+"."+RIGHT$
      (X$,6)
11060 PRINT# 1,X$;
11070 RETURN
```

In dem Unterprogramm werden zuerst fehlende '0' angehängt, anschließend der Dezimalpunkt durch ein Komma ersetzt und zum Schluß eventuell ein Tausender-Punkt eingefügt sowie die Zahl-Zeichenreihe gedruckt. Damit sieht die Druckzeile wie folgt aus:

```
1900 TE$(ZZ)=RA$+STR$(AN)+" "
     +BZ$+"  "
1910 X = EP * GOSUB 11000 (ohne
     Zeile 11060)
1920 TE$(ZZ)=TE$(ZZ)+X$+" "
1930 X = GP * GOSUB 11000 (ohne
     Zeile 11060)
1940 TE$(ZZ)=TE$(ZZ)+X$
```

wenn die Daten für mehrere Drucke zwischengespeichert werden, und

```
1900 PRINT# 1,RA$;AN,BZ$;"  ";
1910 X = EP * GOSUB 11000
1920 X = EP * GOSUB 11000:
     PRINT# 1
```

wenn die Daten direkt gedruckt werden.

Für jede einzelne Zeile muß dann noch der Zeilenzähler erhöht werden mit

```
1950 ZZ = ZZ + 1
```

und die Summe für den Abschluß gebildet werden:

```
1960 SU = SU + GP.
```

Bei Verwendung des Bestandes in der Lagerverwaltung ist noch zu ergänzen:

```
1961 BE = BE - AN
1962 RECORD# 2,(AR)
1963 PRINT# 2,BZ$,EP,BE,...
```

Mit

```
1970 INPUT"Weitere Position";JN$
1980 IF JN$="J" OR JN$ = "j" THEN
     1800
1990 IF JN$="N" OR JN$ = "n"
THEN 2000
1995 GOTO 1970
```

können dann entweder weitere Positionen erfaßt werden oder die Rechnung abgeschlossen werden. Der Rechnungsabschluß wird gebildet durch einen Abschluß-Strich.

```
2000 Schließe Datei 'Artikel'
2005 PRINT# 1,RA$;LEFT$(SU$,79-
LEN(RA$))
2010 ZZ = ZZ - 1
```

der Zwischensumme

```
2020 PRINT# 1,SPC(67-LEN(RA$));
2030 X = SU
2040 GOSUB 11000
2050 ZZ = ZZ - 1
```

und (je nach Bedarf) subtrahieren

A$	— temporäre Hilfsvariable
AN	— Anzahl der Artikel die fakturiert werden
AR	— Artikelnummer
AZ	— Anzahl der Kunden
BE	— Bestand des Artikels
BE$	— Artikelbezeichnung
DA$	— Tagesdatum
DB$	* — Steuerzeichen für 'Drucker auf Breitschrift'
DR$	* — Steuerzeichen für 'Drucker auf Reverseschrift'
EP	— Einzelpreis eines Artikels
EK$	— eingelesene Kundennummer
GP	— Gesamtpreis (jeweils AN * EP)
I	— Laufvariable
JN$	— Abfragevariable für die WERTE 'J' (JA) und 'N' (NEIN)
KN	— Kundennummer (numerisch)
KN$	— Kundenummer (alphanumerisch)
KO	— Kontostand des Kunden
LR$	— Datum der letzten Rechnung des Kunden
NA$	— Kundenname
NF	— nächster freier Datensatz in der OPOS-Datei
PL$	— Postleitzahl des Kunden
O$	— Ort des Kunden
R$	— Hilfsvariable zum Einlesen eines ganzen Datensatzes
RA	— Rabatt
RA$	— Zeichenreihe aus Leerzeichen zum Verschieben des linken Randes
RN	— aktuelle Rechnungsnummer
S$	— Straße des Kunden
SU	— jeweilige Rechnungssumme
SU$	— Unterstreichungsstrich
TE$(72)	— Textarray für die zu druckenden Zeilen
UM	— Umsatz des Kunden
VN$	— Vorname des Kunden
X	— Hilfsvariable zur kaufm. Zahlendarstellung
X$	— Hilfsvariable zur kaufm. Zahlendarstellung
ZZ	— Zeilenzähler

Bild 2. Übersicht verwendeter Variablen (alphabetische Reihenfolge)

des Kundenrabattes (RA), der entweder je Rechnung mit INPUT erfaßt werden kann, oder Bestandteil des Kundendatensatzes ist.

```
2060      PRINT# 1,RA$,"Abzüglich
";RA;" % Rabatt      ";
2070 X = SU - SU / 100 * RA
2080 GOSUB 11000
2090 ZZ = ZZ -1
```

einem weiteren Trennungsstrich

```
2100 PRINT# 1,RA$;LEFT$(SU$,
     79-LEN(RA$))
2110 ZZ = ZZ - 1
```

einem Zwischenergebnis

```
2120 PRINT# 1,SPC(67-LEN(RA$));
2130 X = SU - X : SU = X
2140 GOSUB 11000
2150 ZZ = ZZ - 1
```

wobei in X noch der Wert des Rabattes enthalten ist. Als nächstes muß die Mehrwertsteuer errechnet und gedruckt werden mit

```
2160 PRINT# 1,RA$,"Zuzüglich 13%
     MWST      ";
2170 X = SU - SU * 0.13
2180 GOSUB 11000
2190 ZZ = ZZ - 1
```

noch ein Trennungsstrich

```
2200 PRINT# 1,RA$;LEFT$(SU$,79-
LEN(RA$))
2210 ZZ = ZZ - 1
```

und dem Endergebnis

```
2220 PRINT# 1,SPC(67-LEN(RA$));
2230 X = SU - X : SU = X
2240 GOSUB 11000
2250 ZZ = ZZ - 1
```

Nach zwei (oder mehr) Leerzeilen kann man noch Zahlungsbedingungen ausgeben wie etwa

```
2300 PRINT# 1,RA$;"Zahlbar inner-
halb 10 Tagen mit 2% ";
2310 PRINT# 1,Skonto = DM ";
2320 X = SU - SU * 0.02
2330 GOSUB 11000
2330 ZZ = ZZ - 1
```

oder

```
2300 PRINT# 1,RA$;''Da die Liefe-
rung per NN erfolgt, haben'';
2310 PRINT# 1, "wir 3% Skonto ab-
gezogen = DM ";
2320 X = SU - SU * 0.03
2330 GOSUB 11000
2330 ZZ = ZZ - 1
```

Hier wurde bewußt der Zeilenzähler (ZZ) bei jeder Zeile mitgeführt und nicht die gedruckten Zeilen als gesamtes abgezogen, um die Änderbarkeit des Programms zu erleichtern. Werden keine Fußzeilen (zum Beispiel Bankverbindungen) gedruckt, so kann der Druck des Rechnungsformulares an dieser Stelle abgeschlossen werden mit

```
2400 FOR I = 72 - ZZ TO 72
2410 PRINT# 1
2420 NEXT
```

Die Formulierung mit den beiden '72' in der Schleifenbedingung wurde auch wegen der leichten Änderbarkeit gewählt. Werden andere Papierlängen oder Drucker mit 8 Zeilen/Zoll verwendet, so sind im Programm nur die Konstanten '72' zu ersetzen.

Die Fakturier-Daten lassen sich gut weiterverwenden

Damit ist die engere Fakturierung abgeschlossen, die Druckdatei kann geschlossen werden

```
2430 CLOSE1
```

und die erhaltenen Daten können je nach Bedarf weiterverarbeitet werden. Zum Beispiel in der Kundendatei:

```
3000 KO = KO - SU
```

für den Kontostand des Kunden

```
3010 UM = UM + SU
```

für seinen Umsatz

```
3020 LR$ = DA$
```

als Datum der letzten Rechnung, um spätere 'Karteileichen' automatisch entfernen zu können. Wurden Kundendaten geändert, so darf das Abspeichern nicht vergessen werden. Da die Kundendatei bereits geschlossen wurde, ist sie zuerst wieder zu öffnen:

```
3030 Öffne Datei Kunden auf Kanal
     3
3040 Suche Kunden oder nehme
Datensatznummer wie oben beim
Suchen erhalten
3050 RECORD# 3,(KN)
3060 PRINT# 3,NA$,VN$,S$,...,
     KO,UM,LR$,...
3070 Schließe Datei 'Kunden'.
```

Ebenso können die Daten in eine Offene-Posten-Liste übertragen werden, für die sich auch eine Direktzugriffsdatei empfiehlt. Wir wollen hier annehmen, daß der nächste freie Datensatz in der Offene-Posten-Datei in der Variablen NF abgelegt ist. Es empfiehlt sich, diese Variable mit RN zusammen in der Datei 'Rechnungsnummer' abzulegen, und gegebenenfalls diese Datei von der Offene-Posten-Verwaltung zu aktualisieren. Zeile 1110 würde dann lauten:

```
1110 INPUT# 1,RN,NF
```

und das Abspeichern in der Offenen-Posten-Datei mit den Daten

— Rechnungsnummer (Suchkriterium)
— Kundennummer
— Kundenname
— Datum
— Betrag
— Mahnstufe (anfangs '0')

```
4000 Öffne Datei 'OPOS' auf Kanal 3
4010 RECORD# 3,(NF)
4020 PRINT# 3,RN,KN$,NA$,DA$,
     SU,0
4030 Schließe Datei 'OPOS''
4040 NF = NF + 1.
```

Die obigen Daten sind das Mindeste, was in einer OPOS-Datei gespeichert sein muß, um ein Mahnwesen daraus ableiten zu können. Auf die Problematik der Vertreterbearbeitung soll hier nicht eingegangen werden.

Als letzter Arbeitsgang sind noch die Variablen RN und NF mit ihrem aktuellen Wert abzuspeichern:

```
5000 Öffne Datei 'Rechnungsnum-
mer' auf Kanal 3
5010 PRINT# 3,RN,NF
5020 Schließe Datei 'Rechnungs-
nummer'.
```

Wie man mit etwas mehr Aufwand für den Bildschirmaufbau eine Fakturierung entwickeln kann, zeigt die Abbildung am Beispiel der Fakturierung eines Fotolabors (vergleiche CP Heft 8/82). Solche Feinheiten sind natürlich im allgemeinen Rahmen wie er hier dargestellt wurde nicht möglich, da es zu viele Mikrocomputer mit den unterschiedlichsten Merkmalen gibt.

Aber vielleicht ist es eine kleine Anregung für Sie, das hier Gezeigte an Ihrem Rechner weiter zu verfeinern. (Hans Lorenz Schneider)

— WEBER & SIQUET GmbH — Käfertalerstraße 261-263 — 6800 Mannheim 1 — Telefon 0621/371034 —

OLIVETTI
INK-JET-Drucker PR 2300

INK-JET-Printer auf Graphit-Basis
50 bis 130 Zeilen/Min.
Druckwegoptimierung/Graphikfähigkeit
80, 97 oder 147 Zeichen/Zeile
Parallel-Schnittstelle

Verkaufspreis
DM 1.720,—
inkl. MwSt.

— WEBER & SIQUET GmbH — Käfertalerstraße 261-263 — 6800 Mannheim 1 — Telefon 0621/371034 —

DAS NONSTOP-PROGRAMM
VON
TEXAS INSTRUMENTS

Das Allround-Genie für nur DM 588,-

Entscheiden Sie sich von Anfang an für das richtige System – für den TI 99/4A von Texas Instruments.

Wo andere Home-Computer längst passen müssen, kommt der TI 99/4A erst so richtig in Fahrt.
Die vielfältigen Anwendungsmöglichkeiten, das riesige Software-Angebot und die enorme Speicherkapazität machen den TI 99/4A immer wieder zu einem einzigartigen Erlebnis.
Das bedeutet Spaß, Spannung, Bildung und Wissen für die ganze Familie.
Unsere Programme machen es Ihnen leicht, für jeden das richtige zu finden, z.B.

Parsec*	DM 118,-
Extended Basic	DM 299,-
Multiplan	DM 498,-
Hallenfußball*	DM 89,-
Tombstone City*	DM 89,-
Video-Spiele II	DM 89,-
Invaders*	DM 69,-
Rechenkünstler	DM 62,-
Schachmeister	DM 118,-
Sprach-Synthesizer	DM 298,-
Fernbedienung	DM 98,-
Computer-Konsole	DM 588,-

* Nur in Verbindung mit Fernbedienung

KÜBLER
BÜRO+DATENTECHNIK

Erleben Sie den TI 99/4 A jetzt in Ihrem Fachgeschäft – er wird auch Sie auf Anhieb begeistern.
Alle Computerfreunde, die keine Gelegenheit haben, sich über unser reichhaltiges Angebot in unserem Hause zu informieren, bedienen wir gerne per Post. Schicken Sie einfach den Bestellschein an:

Fa. Kübler · Abt. VS · Silberburgstraße 145 · 7000 Stuttgart 1 · Tel. 07 11/61 06 51

BESTELLSCHEIN Ich bestelle per Nachnahme ☐ V-Scheck anbei ☐

Artikel	Preis DM	Anzahl	Gesamtpreis
TI 99/4A Home Computer	**588,-**		

Bitte in Blockschrift ausfüllen:

Name/Vorname

Straße/Haus-Nr.

Postleitzahl/Ort Telefon

CP 2/83

Der TI-Professional-Computer mit Supergrafik

Das erste, was am Professional Computer von Texas Instruments auffällt: Er ist ganz konsequent nach dem gleichen Konzept gebaut, wie der IBM Personal Computer. Das zeigt sich nicht nur bei der Hardware, sondern auch bei der Werbung. Die TI-Anzeigen für den Professional in den USA sehen den IBM-Anzeigen sehr ähnlich. So wie in allen IBM-Anzeigen eine rote Rose zu sehen ist, versucht TI mit einer grünen Lampe ein Erkennungsmerkmal zu etablieren. Was das bedeuten soll, scheinen allerdings nur sie selbst zu wissen. Oder soll da irgendwem ein Licht aufgehen?

Man kann jedoch beim TI Professional Computer auf keinen Fall von einem Nachbau sprechen. Das würde auch nicht ganz zum Bild eines der größten Halbleiterhersteller passen. Einige Eigenschaften wurden gegenüber dem Vorbild verändert und teilweise sogar verbessert. Die Begründung von Texas Instruments für die Anlehnung an das IBM-Konzept lautet, daß zum Beispiel in der Automobilindustrie auch jeder Hersteller seine Motoren so konstruiert, daß der an allen Tankstellen verfügbare Treibstoff eingesetzt werden kann. Oder auf die EDV umgesetzt: Auch ein Computer sollte sich an die Standards halten, die sich im Markt entwickelt haben. Nur so steht ihm gleich von Anfang an ein ausreichendes Softwareangebot zur Verfügung. Der TI Professional ist also nicht als reines Konkurrenzprodukt zum IBM-PC gedacht, sondern Texas Instruments möchte damit von der Tatsache profitieren, daß für diesen Rechner ein großes und ständig wachsendes Softwareangebot verfügbar ist. Die Entscheidung für diesen Schritt wird Texas Instruments sicher nicht leicht gefallen sein, denn so mußte auch auf den Einsatz eines eigenen Mikroprozessors verzichtet werden, ein Novum in der Firmengeschichte.

Der TI Professional besteht aus drei Komponenten: der Systemeinheit, der Tastatur und einem Bildschirm. Das Herz der Systemeinheit ist ein 8088-Prozessor, der mit 5 MHz Taktfrequenz betrieben wird. Wie beim IBM-PC ist noch Platz für den Numerikprozessor 8087 vorgesehen. In der Systemeinheit ist Platz für zwei Diskettenlaufwerke und fünf Erweiterungskarten. Die Laufwerke von TI haben eine Kapazität von 320 KByte. Anstatt eines zweiten Laufwerks ist optional eine Festplatte mit 5 oder 10 MByte erhältlich. Standardmäßig ist der Pro mit 128 KByte Speicher ausgestattet. Dieser läßt sich derzeit in 64-KByte-Schritten auf 256 KByte ausbauen. Für diese Speichererweiterung ist ein separater Steckplatz vorhanden. Da dieser Steckplatz auf der gleichen Linie liegt, wie ein normaler Erweiterungssteckplatz, können in diesem nur kurze Erweiterungskarten verwendet werden. Die Belegung des Erweiterungsbus entspricht zwar weitgehend der des IBM-PC, es bestehen jedoch ein paar Unterschiede, so daß IBM-Er- ►

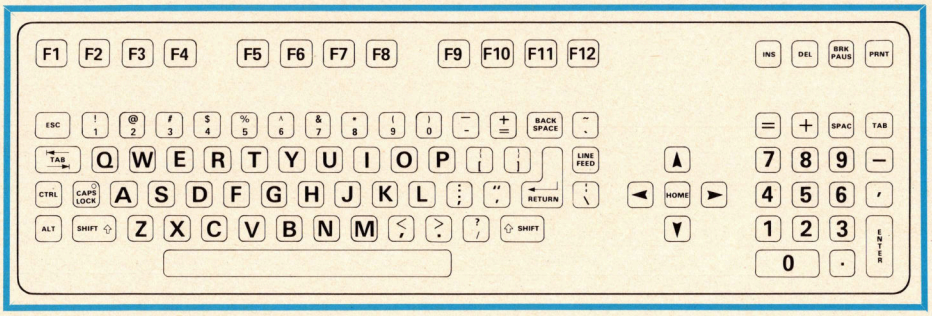


Bild 1. Tastaturdiagramm

weiterungskarten wohl nicht verwendet werden können.

Standardmäßig ist eine Parallel-Schnittstelle (Centronics) eingebaut. Eine synchrone/asynchrone Schnittstelle (RS232C/V 24) ist optional erhältlich.

Ein großer Unterschied besteht beim ROM. Beim IBM-PC sind außer den 8 KByte BIOS (Ein-/Ausgabe-Routinen) noch 32 KByte ROM mit Kassetten-Basic vorhanden. Der TI Professional aber ist eine reine Diskettenmaschine. Er hat also nur 8 KByte ROM, in denen sich das BIOS und Testroutinen befinden.

Die Tastatur ist in drei Bereiche eingeteilt; links eine Schreibmaschinentastatur mit 12 Funktionstasten, rechts eine Zahlentastatur mit Einfüge- und Lösch-Tasten und dazwischen fünf Tasten für die Cursorbewegung (Bild 1). Insgesamt sind das 97 Tasten. Die Aufteilung ist besser als beim IBM-PC; die Funktionstasten befinden sich über der Schreibmaschinentastatur und korrespondieren mit der Anzeige auf dem Bildschirm. Die Zahlentastatur und die Cursorsteuerung sind separat, so daß die Bedienung etwas einfacher ist als auf dem Vorbild. Die Taste Caps-Lock hat eine eingebaute LED. Doch auch diese Tastatur ist nicht ganz makellos; die Tasten INS, DEL, PAUS und PRNT sind über der Zahlentastatur angeordnet, obwohl sie logisch über die Cursortasten gehören würden. Diese Anordnung hat auch den Nachteil, daß diese Tasten zu weit außerhalb des normalen Schreibbereiches liegen. Der Neigungswinkel der Tastatur ist im Bereich von 5 bis 15 Grad verstellbar.

Die Bildschirmsteuerkarte kann einen monochromen oder einen Farbbildschirm steuern. In der Grundausführung verfügt sie über den nötigen Speicher für 25 Zeilen mit 80 Zeichen. Die Karte ist jedoch schon für die Erweiterung zur Farbgrafikkarte vorgesehen. Diese Erweiterungskarte wird auf die Bildschirmsteuerkarte aufgesteckt. Damit wird eine Auflösung von 720 x 300 Punkten erzielt. Die Anzahl der Farben hängt von der Ausbaustufe ab. Man kann wählen zwischen einer und drei Grafikebenen. Bei einer Ebene sind zwei Farben möglich, also einfache Schwarzweiß-Grafik, bei drei Ebenen können acht Farben dargestellt werden. Ob die Darstellung in Farbe erfolgt oder nicht, hängt dann allein vom Monitor ab. Auf einem monochromen Bildschirm werden Grafiken in verschiedenen Graustufen dargestellt. Dieser Bildschirm hat eine Diagonale von 12 Zoll, der Farbmonitor ist mit 13 Zoll ein wenig größer. Einzelne Zeichen werden mit 7 x 9 Punkten in einer 9 x 12-Matrix angezeigt. Der Zeichengenerator besteht aus einem 4 KByte ROM, so daß 256 Zeichen verfügbar sind. Es ist Platz für ein weiteres ROM/EPROM vorhanden, mit dem der Zeichensatz auf 512 Zeichen er-

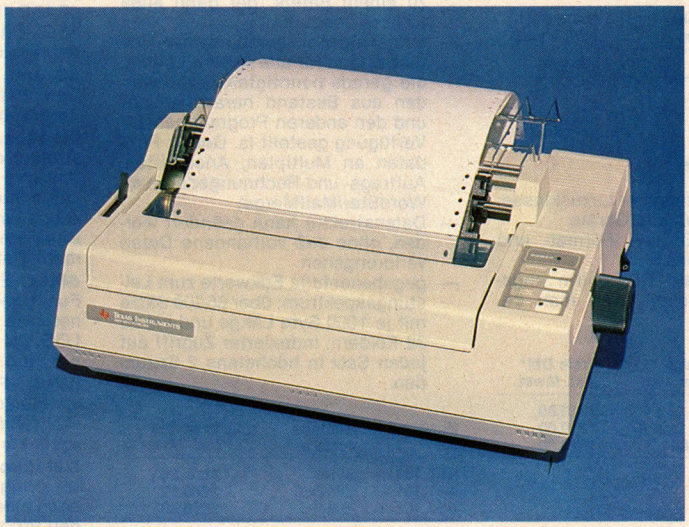
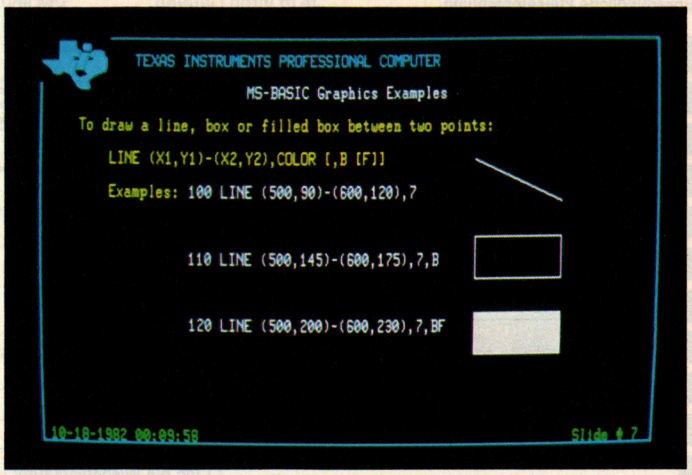


Bild 2. Der Drucker 850, der kompatibel zum Epson MX-80 sein soll



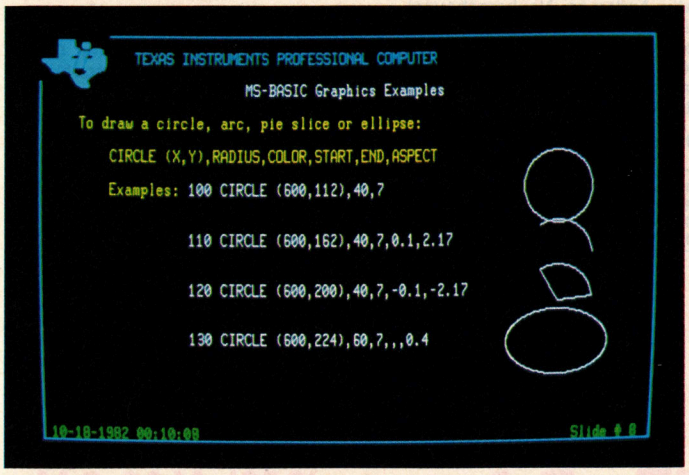
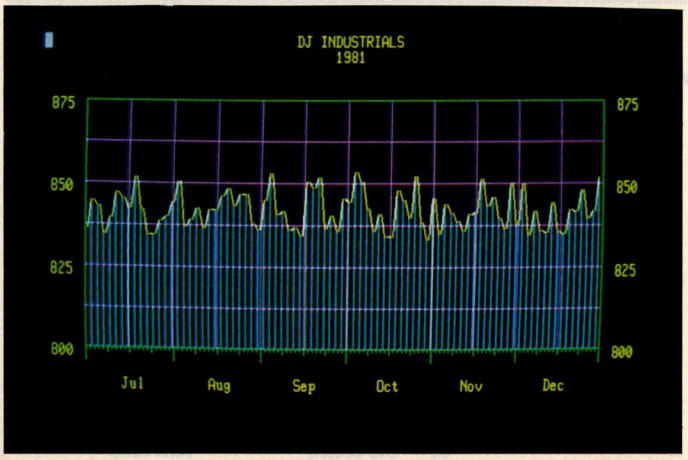
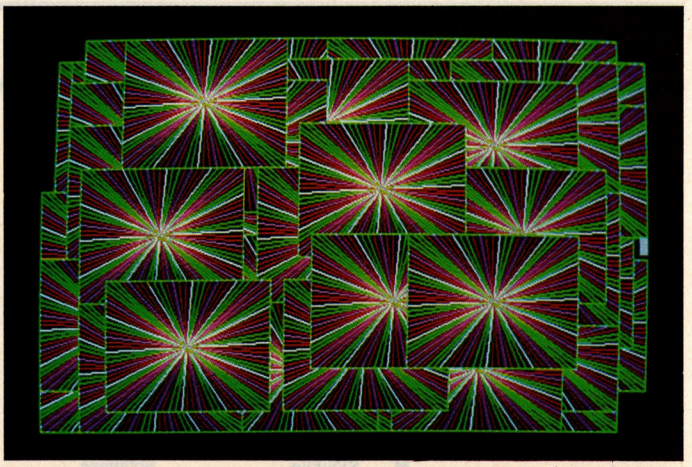


Bild 3 und 4. Hier erläutert ein Demoprogramm die Basic-Befehle LINE und CIRCLE





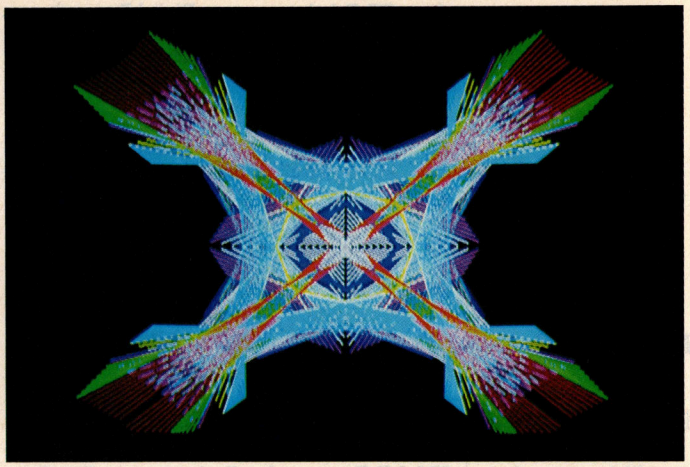


Bild 7 bis 10. Dies sind Beispiele für die exzellenten Farbgrafiken des Professional Computers

weitert werden kann. Der Zeichensatz ist der gleiche wie beim IBM Personal Computer. Die Zeichen können revers, unterstrichen, blinkend oder in verschiedenen Intensitäten beziehungsweise Farben dargestellt werden. Es sind acht Farbtöne möglich: Schwarz, Blau, Rot, Magenta, Grün, Zyan, Gelb und Weiß. Für die Farbgrafik werden drei 32 KByte große Speicherblöcke verwendet, von denen aber nur jeweils 27600 Bytes benutzt werden.

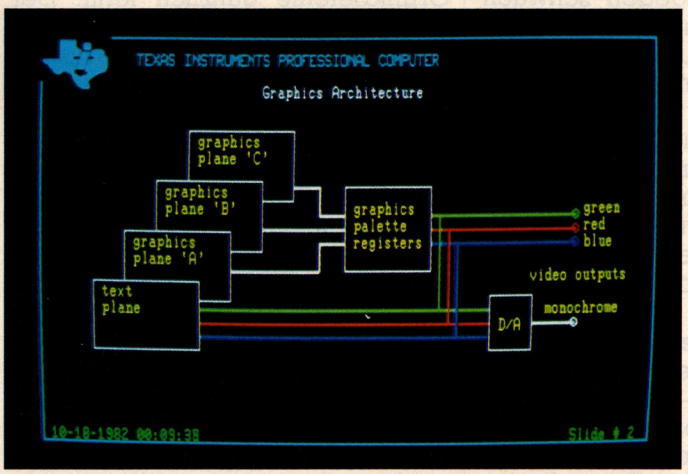
TI bietet zum Professional verschiedene Drucker an. Besonders interessant ist das Modell 850 (Bild 2). Dieser Drucker ist weitgehend kompatibel zum Epson MX-80, der auch von IBM verwendet wird. Software, die vom PC her umgestellt wurde, braucht also nicht auf einen Drucker angepaßt werden.

Für den Test stand ein Rechner mit 256 KByte, zwei Diskettenlaufwerken, Grafikcontroller und Farbmonitor zur Verfügung. Mit dieser Konfiguration entstanden auch die Grafiken, die auf diesen Seiten zu sehen sind.

In Deutschland wird derzeit die amerikanische Version des Professional Computers im Direktvertrieb nur an Großkunden verkauft. Der Vertrieb über ein Händlernetz an Endkunden soll ab dem vierten Quartal erfolgen, wenn Tastatur, Zeichensatz und Dokumentation an deutsche Verhältnisse angepaßt beziehungsweise übersetzt sind.

Auch bei den Betriebssystemen folgt Texas Instruments dem Weg, den IBM vorgegeben hat. Das Haupt-Betriebssystem ist MS-DOS, optional sind auch CP/M-86 und das UCSD p-System erhältlich. MS-DOS läuft derzeit in der Version 1.25, die Version 2.00 wird voraussichtlich ab Herbst erhältlich sein, so daß in Deutschland wahrscheinlich alle Systeme damit ausgeliefert werden.

In bezug auf die Verfügbarkeit von Software scheint TI jedoch we-



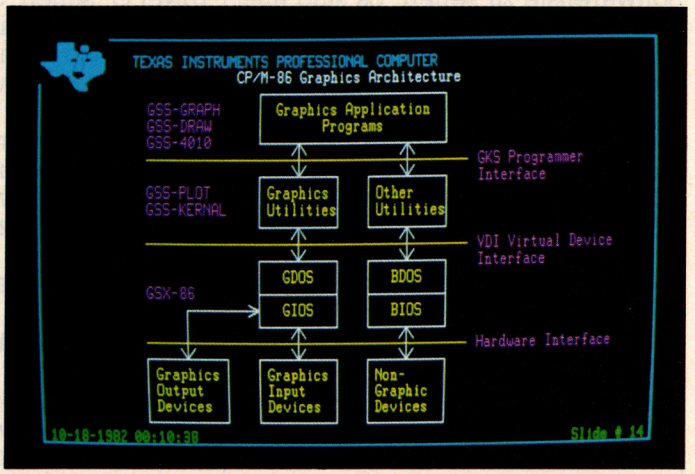


Bild 5 und 6. Der Aufbau der Grafikhardware (links) und die Grafikerweiterungen des Betriebssystems CP/M-86 (rechts)

Anwendung	DOS	Programm	Anbieter
Kommunikation	M/C	ASCOM86	Lifeboat
	M	3780 Communications	TI
	M	TTY Communications	TI
Datenbanken und Dateiverwaltungen	C	Access Manager	Digital Research
	M/C	DataStar	MicroPro
	M/C	dBase II	Ashton Tate
	C	Display Manager	Digital Research
	M	Easyfiler	IUS
	M	Formula	Lifeboat
	M/C	Infostar	MicroPro
	M	PFS:File	Software Publishing
	M	PFS:Report	Software Publishing
	M/C	Super Data	Sorcim
	M/C	Super Sort	MicroPro
Finanzplanung und Tabellenkalkulation	M/C	Bottom Line Start.	Ashton-Tate
	M/C	CalcStar	MicroPro
	M	EasyPlanner	IUS
	M	Multiplan	Microsoft
	M/C	PeachCalc	Peachtree
	M/C	SuperCalc	Sorcim
	M/C	SuperCalc 2	Sorcim
	M/C	Financial Planner	Ashton Tate
Grafik	M/C	Business Graphics	Peachtree
	C	GSX	Digital Research
	M/C	Graphics Language	Peachtree
	M	PFS:Graph	Software Publishing
	M/C	Superchart	Sorcim
	U	Turtlegraphics	SofTech
Textverarbeitung	M	Easywriter	IUS
	M/C	Mail Merge	MicroPro
	M/C	PMATE-86	Lifeboat
	M/C	SuperWriter	Sorcim
	M/C	WordStar	MicroPro
Programmiersprachen	M	MACRO-86	Microsoft
	C	CBASIC-86	Digital Research
	M	MS-BASIC	Microsoft
	M	Lattice C	Lifeboat
	C	CIS-Cobol	Digital Research
	C	Level II Cobol	Digital Research
	M	MS-Cobol	Microsoft
	M	RM-Cobol	Ryan-McFarland
	M	MS-Fortran	Microsoft
	C	Pascal/MT+ 86	Digital Research
	M	MS-Pascal	Microsoft
	U	UCSD-Pascal	SofTech
Utilities	M	Emulator/86	Lifeboat
	M	EM80/86	Lifeboat
	M	Fabs	Lifeboat
	M	Panel-86	Lifeboat
	M	UT86	Lifeboat

Tabelle 1. Ein Auszug aus der Softwareliste für den TI Professional. Unter DOS bedeutet M = MS-DOS, C = CP/M-86 und U = USCD p-System.

nig von seinem Vorbild gehalten zu haben. Während IBM im Vertrauen auf den Markt kaum Software anbietet und dies Softwarehäusern überläßt (mit Erfolg), hat TI schon lange vor der offiziellen Ankündigung amerikanischen Softwareherstellern Professional Computer zur Verfügung gestellt, so daß gleich von Anfang an die wichtigsten Softwareprodukte erhältlich waren. Dazu gehört auch eine Hard-/Software-Kombination, nämlich die Z80-Karte von Xedex, die auf dem IBM-PC unter dem Namen Baby Blue bekannt geworden ist und hier sinnigerweise Baby Tex heißt. Damit kann der größte Teil der Programme aus der 8-Bit-CP/M-Welt auf dem Professional laufen. In Tabelle 1 ist ein Auszug aus der TI-Softwareliste zu sehen. Es sind alle wichtigen Standardpakete verfügbar und es werden die wichtigsten Anwendungsgebiete abgedeckt. Bei den Programmiersprachen ist eine besonders zu erwähnen, nämlich RM-COBOL. Es soll laut TI kompatibel zu dem Cobol auf ihren BS-Systemen sein (und zu dem auf TI-Rechnern, die von anderen Anbietern vertrieben werden, zum Beispiel Kienzle und Triumph-Adler); ein wichtiger Punkt für Besitzer solcher Geräte.

Was unterscheidet nun aber den TI Professional Computer sonst noch von seinem Vorbild dem IBM Personal Computer und welche Gründe könnten dafür sprechen, daß man sich für ihn entscheidet? Da ist natürlich vor allem die Grafik zu nennen. Die Grafikkarte des IBM-PC erlaubt maximal eine Auflösung von 640 x 200 Punkten. Das jedoch nur in Schwarzweiß. Bei farbigen Grafiken sind höchstens 320 x 200 Punkte möglich. Auf einem Bildschirm können vier Farben und eine von 16 Hintergrundfarben verwendet werden, wobei aber nur 40 Zeichen Text in eine Zeile passen. Hier sieht man schon den gravierenden Unterschied. Auf dem Pro steht bei voll ausgebauter Grafikkarte jederzeit eine Grafikauflösung von 720 x 300 Punkten bereit, mit einer Textdarstellung von 80 Zeichen pro Zeile. Das Ergebnis ist auf den Bildschirmfotos (Bilder 7 bis 10) zu sehen. Farbgrafiken des IBM-PC sehen dagegen schwach aus. Darüber hinaus hat der (zur Zeit nur in den USA erhältliche) Farbbildschirm von IBM einen großen Nachteil: Er spiegelt sehr stark. Wenn man hinten eine Lichtquelle hat (Fenster, Lampe), dann ist es sehr schwierig, sich auf die Anzeige zu konzentrieren. Der TI-Farbmonitor ist dagegen entspiegelt und stellt Farben etwas kräftiger dar.

In einigen älteren amerikanischen Tests war der TI etwas langsamer als der IBM-PC. Mittlerweile sind die Betriebssystemsoftware und das Basic überarbeitet worden und er war bei allen Tests die ich durchgeführt habe sogar schneller.

Auch in anderen Punkten schneidet der Pro im Vergleich mit dem IBM-PC gut ab. Bei einem Professional mit Farbgrafik bleiben drei Erweiterungssteckplätze frei. Beim IBM ist es bei vergleichbarem Ausbau nur einer. IBM liegt allerdings in einem Punkt ganz klar vorn: Die Dokumentation des Professional ist nicht immer so gut wie die (englische) des Vorbilds. So ist zum Beispiel das Basic-Handbuch recht unübersichtlich, da die Erläuterung eines Befehls immer gleich im Anschluß an den vorherigen beginnt und nicht auf einer neuen Seite anfängt. Teilweise haben sich auch echte Fehler eingeschlichen. So ist zum Beispiel ein kleines Übersichtsbüchlein beigefügt, in dem allerdings die Beschreibung der Grafikbefehle gänzlich fehlt. Der Wert ist also mehr als zweifelhaft.

Das Basic selbst hält sich bis auf eine Ausnahme — oder besser Ergänzung — vollständig an die Vorgabe des IBM-PC. Diese Ergänzung ist ein Befehl mit dem Namen PALETTE. Damit können im Bruchteil einer Sekunde alle Punkte einer Farbe auf dem Bildschirm in eine andere Farbe verändert werden. Das erlaubt die leichte Erzeugung von Bewegungseffekten. Das Bild mit dem Titel »Worlds«, das im Inhaltsverzeichnis zu sehen ist, wirkte zum Beispiel so, als ob sich die einzelnen Kugeln drehen wür-

Fortsetzung auf Seite 155

Fortsetzung von Seite 140

den. Basic-Programme vom IBM-PC können problemlos geladen werden, wenn jedoch Grafikbefehle verwendet werden, erscheinen die Grafiken auf dem TI-Bildschirm in die linke obere Ecke gequetscht und eventuell dazugehörige Bezeichnungen über den Bildschirm verstreut, da die Grafikauflösungen einfach zu unterschiedlich sind.

Der Preis für die Grundkonfiguration mit 128 KByte Speicher, zwei Diskettenlaufwerken (je 320 KByte) und einem monochromen Monitor beträgt 10450 Mark. Das liegt geringfügig unter dem Preis des IBM-PC für eine vergleichbare Konfiguration. Die synchrone/asynchrone Schnittstelle kostet 795 Mark, der 850-Drucker 1690 Mark, der TI-Farbmonitor 2500 Mark, die Grafikkarte mit einer Ebene 950 und mit drei Ebenen 1850 Mark (alle Preise zuzüglich Mehrwertsteuer). Die Farbgrafikkarte des IBM-PC kostet im Vergleich 885 Mark. Der Preis für die TI-Grafikkarte sieht also etwas hoch aus; das ist aber keineswegs so, denn man bekommt hier eine wesentlich höhere Auflösung mit entsprechend viel Speicher (bei drei Ebenen 96 KByte). Der Texas Instruments Professional Computer ist also auf jeden Fall immer dann vorzuziehen, wenn es um gute Farbgrafiken geht. Doch auch in fast allen anderen Punkten kann er mit seinem Vorbild mithalten.

Zur Verbesserung der Benutzerschnittstelle ist für Ende des Jahres eine interessante Erweiterung angekündigt: Spracherkennung, -aufzeichnung und -wiedergabe. Mit Hilfe eines schnellen 32-Bit Signal-Prozessors (TMS 320) können dem Professional Computer dann 40 bis 50 Phrasen antrainiert werden, die er dann versteht und mit der Ausführung von beliebig zugeordneten Kommandos quittiert. Die Hardware dafür beansprucht einen Steckplatz. Damit ist auch digitale Sprachaufzeichnung möglich; es können pro 320-KByte-Diskette ungefähr 16 Minuten Sprache gespeichert werden. Dazu gehört Software zur Bearbeitung; unter anderem auch ein Programm, daß den Einsatz des Professional als Telefonanrufbeantworter ermöglicht. Der Einsatz dieses Programms hängt allerdings noch von der Erlaubnis der Post ab. Ein Prototyp dieses Moduls war auf der Hannover-Messe zu sehen. Der Preis steht noch nicht fest. (jü)

Inserenten in dieser Ausgabe

aaa	123
Adcomp	21
Anderson Jacobsen	33
Arhelger	27
Basic Bär	42
Bavico	43
Brandner	58
Brockhaus + Müller	110
Burkhardt	97
C.V.B. Buerschaper	27
CE	25
City Elektronik	5
Commodore	27,29
Computer Buchladen	126-129
Computer Forum	42
Computer Modular	117
Computerladen	97
Computronic	117
Corvallis	111
Createam	117
CS Computer Software	115
Data Becker	19
Datasoft	121
Deutsche Olivetti	160
Digital Equipment	14,15
Digital Research	35
Dragon	12,13
Dynatrade	43
EBS	105
Elcon	111
Füssner	63
GMB Soft	121
Görlitz	97
Grässer	58,63
Graumann	117
Hähner	58
Hees	69
Heilmeier + Steinberger	113
Hirschberg	58
Hofacker	41
Holtkötter	55
Hübner + Worm	50
I + B	58
IBBG	121
ibi	112
Inetec	43
IWT	51
Jodlbauer	58
Jöllenbeck	2
M&T Software Verlag	67,125,136,137
Mamaco	42
Maxell	105
Micro Versand	51
Microdex	63
Micropro	71
Microscan	119
Mirwald	63
Mitsui	99
Memorex	45
MMOS	45
D.Müller	58
M.Müller	112
NBN	37
NEC	51
Pflaum	105
Pro Computer	121
Reis	58
Renner	95
Rösch	123
Schneider	115
Schüngel	123
SE-Spezial-Elektronik	22,23
Sirius	109
SM Software	39,102,103
Speech	112
Stemmos	123
Sybex	52
te-wi	114
TecSys	112
Timesoft	105
Toshiba	32
Vieweg	45
Waldsee	115
Wemper	121

Impressum

Herausgeber: C.F.von Quadt, Otmar Weber

Chefredakteur: Michael M. Pauly (py)
Stellv. Chefredakteur: Michael Scharfenberger (sc)
Redakteure: Silvia Gutschmidt (gu), Christian Jansen (ja), Günter Jürgensmeier (jü), Johannes Leckebusch (le), Manfred S. Schmidt (mss)
Redaktionsassistenz: Dagmar Zednik (237)
Layout: Alexander Gerhardt
Fotografie: Janos Feitser, Titelfoto: Alex Kempkens

Auslandsrepräsentation: Schweiz: Markt&Technik-Agentur, Postfach, CH-8033 Zürich, Tel. 01-362 1320

Manuskripteinsendungen: Manuskripte werden gerne von der Redaktion angenommen. Honorare nach Vereinbarung. Die Zustimmung des Verfassers zum Abdruck wird vorausgesetzt. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

Herstellung: Klaus Buck (180), Leo Eder (181)

Anzeigenleitung: Peter Schrödel (156); Anzeigenverkauf: Kurt Ebner (211), Helmut Kunkel (151), Waltraud Werner (236), Einkaufsführer: Marion Heinrichs (118)

Anzeigenverwaltung und Disposition: Patricia Schiede (172)

Anzeigenformate: 1/1-Seite ist 266 Millimeter hoch und 185 Millimeter breit (3 Spalten à 58 mm oder 4 Spalten à 43 Millimeter). Vollformat 297x210 Millimeter. Beilagen und Beihefter siehe Anzeigenpreisliste.

Anzeigenpreise: Es gilt die Anzeigenpreisliste Nr.2 vom 1. Januar 1983.
1/1 Seite sw: DM 3960,-. Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala DM 720,-. Vierfarbzuschlag DM 1860,-.
Mindestgröße Produktanzeigen = 1/8 Seite = DM 495,-. Kleinanzeigen im CP-Markt: je Druckzeile und Spalte (Spaltenbreite 43 mm) gewerblich DM 7,- (+ MwSt.), privat DM 5,- (inkl. MwSt.).
CP-Einkaufsführer: Mindesthöhe 15 mm, Mindestabnahme 6 Eintragungen; Spaltenbreite 58 mm, je mm Abdruckhöhe DM 6,-. Chiffregebühr: DM 10,- (Inland), die gesetzliche MwSt. wird jeweils zugerechnet.

Vertriebsleitung, Werbung: Hans Hörl

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Verlagsunion Zeitschriften-Vertriebs-KG, Friedrich-Bergius-Str. 7, 6200 Wiesbaden, Telefon 06121/266-0.

Erscheinungsweise: »Computer persönlich« erscheint 14täglich, Mittwoch.

Bezugsmöglichkeiten: Abonnenten-Service: Silvia Augustin Tel. 089/4613238. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen. Das Abonnement verlängert sich um ein Jahr, wenn es nicht zwei Monate vor Ablauf gekündigt wird.

Bezugspreise: Das Einzelheft kostet DM 4,-. Der Abonnementspreis beträgt im Inland DM 78,- pro Jahr für 26 Ausgaben. Darin enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren. Der Abonnementspreis erhöht sich um DM 25,- für die Zustellung im Ausland, für die Luftpostzustellung in Ländergruppe 1 (z.B. USA) um DM 65,-, in Ländergruppe 2 (z.B. Hongkong) um DM 100,-, in Ländergruppe 3 (z.B. Australien) um DM 130,-.

Druck: R. Oldenbourg, Hürderstr. 4, 8011 Kirchheim
Auch Anschrift für Beihefter und Beilagen.

Urheberrecht: Alle in »Computer persönlich« erschienenen Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten. Reproduktionen gleich welcher Art, ob Fotokopie, Mikrofilm oder Erfassung in Datenverarbeitungsanlagen, nur mit schriftlicher Genehmigung des Verlages. Anfragen sind an Hans Hörl zu richten. Für Schaltungen und Programme, die als Beispiele veröffentlicht werden, können wir weder Gewähr noch irgendwelche Haftung übernehmen. Aus der Veröffentlichung kann nicht geschlossen werden, daß die beschriebenen Lösungen oder verwendeten Bezeichnungen frei von gewerblichen Schutzrechten sind.

© 1983 Markt & Technik Verlagsgesellschaft mbH, Redaktion »Computer persönlich«.

Verantwortlich: Für redaktionellen Teil: Michael M. Pauly. Für Anzeigen: Peter Schrödel.

Geschäftsführer: C.F.v.Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt&Technik Verlagsgesellschaft mbH, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon 089/4613-0, Telex 05-22052

ISSN 0722-0987

Telefon-Durchwahl im Verlag:
Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089-4613 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.
Info CP in ***Düsseldorf*** und ***Berlin***
☎ ★ 15982 # (Bildschirmtext)

Das Nonstop-Programm.

Entscheiden Sie sich von Anfang an für das richtige System – für den TI 99/4A von Texas Instruments.

Wo andere Home-Computer längst passen müssen, kommt der TI 99/4A erst so richtig in Fahrt.
Er ist einer der derzeit fortschrittlichsten und flexibelsten Home-Computer auf dem Markt. Seine vielfältigen Anwendungsmöglichkeiten, das riesige Software-Angebot, meist in deutsch, die enorme Speicherkapazität von 16 Kbyte in der Grundausstattung und die exakt aufeinander abgestimmten Ausbaukomponenten machen ihn zu einem einzigartigen Rund-um-die-Uhr-Erlebnis.
Das bedeutet Spaß, Spannung, Bildung und Wissen für die ganze Familie.

Spielen und lernen in neuen Dimensionen mit dem riesigen Software-Angebot von Texas Instruments, davon viele als Solid State Software™ in einfacher Steckmodultechnik: einlegen – loslegen.

In den folgenden Bereichen gibt es viele spannende und interessante Programme:

- **Aus- und Weiterbildung**
- **Creative Unterhaltung**
- **Organisation des Heimbereichs**
- **Lernen von Programmiersprachen** (Basic, Assembler, Logo, Pascal)

Erleben Sie den TI 99/4A jetzt in Ihrem Fachgeschäft, er wird auch Sie auf Anhieb begeistern.
Mehr als 1 Mio. verkaufte Home-Computer von Texas Instruments!

TI 99/4A von Texas Instruments – es gibt keinen vielseitigeren.

TEXAS INSTRUMENTS

Bild 1. TI 99/4A mit Fernsehgerät, Kassettenrecorder, System-Erweiterungsbox und Joysticks

TI 99/4A: ein Heimcomputer, der eine Menge kann

Nach den Vorstellungen von Texas Instruments sollen Heimcomputer, wie der 99/4A in Zukunft im Haushalt eine ähnliche Rolle spielen wie heute ein Plattenspieler. Inwieweit sind heute, Jahre nach der Einführung dieser Geräte, die Voraussetzungen dafür geschaffen?

Die preislichen Voraussetzungen sind auf jeden Fall erfüllt: 1979 mußte man für den TI 99/4 noch zirka 2700 Mark bezahlen (mit Monitor). Bereits Ende 1980 fiel der Preis unter 1500 Mark. Im Sommer '82 wurde der TI 99/4A für 1248 Mark angeboten und heute gibt es ihn für zirka 550 Mark. Das ist für ein 16-Bit-System ein niedriger Preis, an dem kaum noch viel verdient werden kann — im Gegenteil: Man munkelt, daß man bei diesen Preisen schon in der Verlustzone sei (keine Angst: Verdient wird dann an den Erweiterungen und am Zubehör, wie Software etc.) Der im TI 99/4A eingesetzte Prozessor gehört der TMS-9900-Familie an, die vom Hersteller bereits in größeren Computern erprobt worden ist. »Eine 16-Bit-Zentraleinheit gewährleistet, daß auch in Zukunft für diesen Rechner kompatible Programme entwickelt werden. Während für 8-Bit-Systeme wohl nur mehr wenige neue Programme zu erwarten sind«, meint der Hersteller — eine Aussage, die vielleicht auf Personal Computer zutrifft, im Home-Computerbereich jedoch mit Skepsis zu betrachten ist: Denn alle Homecomputer-Hersteller bieten nur für ihre eigenen Modelle Software an, die auch nur auf diesen lauffähig ist (eine Tatsache, die von den Käufern übrigens als sehr ärgerlich empfunden wird). Da kann man sich dann zu Recht fragen, welche Rolle es spielt ob ein 8- oder 16 Bit-Prozessor eingesetzt wird, wenn man doch nur Software vom jeweiligen Computerhersteller kaufen kann. Fazit: Software-Kompatibilität zwischen den einzelnen Homcomputern verschiedener Anbieter ist zur Zeit noch Illusion.

Wie der TI 99/4A und seine Peripherie überhaupt aussehen, zeigt Bild 1. Die Schreibmaschinentastatur der Konsole (mit US-ASCII-Standard-ähnlicher Tastenanordnung) besteht aus 48 Tasten, die fast alle Zweitfunktionen haben. Der TI 99/4A erlaubt Groß- und Kleinschreibung. An der Frontseite dieses Heim-Computers befindet sich der Eingabeschacht für die Solid-State-Software-Module, auf denen die meisten käuflichen Programme gespeichert sind.

Platzsparende Modul-Technik

Der Vorteil solcher Programmmodule ist, daß keinerlei Kenntnisse über den Computer selbst zum Start eines Programms nötig sind. Der Benutzer steckt das entsprechende Modul in den Eingabeschacht der Konsole, schaltet das Gerät ein und ist mitten im Programm; kein Laden und Starten des Programms ist mehr nötig. Ein weiterer Vorzug der Module: Die Programme müssen nicht in den Arbeitsspeicher geladen werden, der damit frei für beliebige Daten ist — eine Voraussetzung um zum Beispiel mit einem Textverarbeitungsprogramm längere Briefe oder Datenwerte erstellen zu können. Denn die Module haben direkten Zugang zum Betriebssystem und der CPU und belegen damit keinen wertvollen Arbeitsspeicher. Die Kapazität der Module liegt zwischen 6 und 30 KByte ROM. Bild 2 zeigt, wie das Modul »TI Extended Basic« — ein Basic-Interpreter — in den Eingabeschacht gesteckt wird.

Der TI 99/4A selbst verfügt über 26 KByte ROM, der das Betriebssystem und den Basic-Interpreter enthält. Der freiverfügbare Arbeitsspeicher ist 16 KByte groß. Er kann durch ein Mini-Memory-Modul oder eine Erweiterungskarte, die weiter unten beschrieben wird, vergrößert werden. Der 14 KByte-Speicher des Minimoduls besteht aus einem 4 KByte batteriegepuffertem RAM-, 4-KByte-ROM- und 6-

KByte-Grafik-Bereich. Durch die Batteriepufferung der 4-KByte-RAM werden die in diesem Speicherbereich abgelegten Daten auch nach Abschalten des TI 99/4A nicht zerstört. Außerdem stellt es zusätzliche TI-Basic-Unterprogramme sowie ein Fehlersuchprogramm zur Verfügung.

Da der Computer auf 110 Volt (Netzspannung in den USA) ausgerichtet ist, ist ein externer Adapter nötig, der die bei uns übliche Wechselstromspannung von 220 V auf 110 V transformiert. Der Adapter ist ebenso im Lieferumfang enthalten wie ein PAL-Modulator, den man benötigt, um den TI 99/4A an ein Fernsehgerät (möglichst Farbfernsehgerät) anzuschließen. Dieser PAL-Modulator verbindet den Computer mit dem Antennenausgang des Fernsehgeräts. Der TI 99/4A kann Bilder in 16 verschiedenen Farben in einer Auflösung von maximal 192 x 256 Bildpunkten darstellen.

An der Rückseite und der linken Seite der Konsole befindet sich jeweils ein Stecker, an den ein handelsüblicher Audio-Kassettenrecorder angeschlossen werden kann; bei Bedarf also zwei Kassettenrecorder gleichzeitig. Das Kassettenrecorderkabel muß man sich zusätzlich kaufen (bei einem als Zubehör erhältlichen Kassettenrecorder ist es im Lieferumfang bereits enthalten). Bei der Wahl eines Kassettenrecorders sollte man darauf achten, daß er gleichmäßige Bandgeschwindigkeit und gute Wiedergabe gewährleistet. Außerdem ersparen qualitativ hochwertige Kassetten viel Ärger und Mühe beim Speichern und Laden der Daten von Kassetten.

An denselben Stecker, der für den Anschluß eines Kassettenrecorders benutzt wird, werden Joysticks, die häufig in Spielen und Lernprogrammen eingesetzt werden, angeschlossen.

Bereits in seiner einfachsten Ausstattung kann der TI 99/4A eine ganze Menge, da der Anbieter zahlreiche Software-Module anbietet. Dieser Heimcomputer ist vom Preis und Können her geeignet dem Computer-Interessierten den Einstieg in die Welt der Computer zu ermöglichen. Es gibt zahlreiche spannende Spiele, die viel Spaß machen und an denen der geschickte Umgang mit der Tastatur geübt sowie ein Überblick über die grafischen und akustischen Fähigkeiten dieses Computers gewonnen werden kann. Sie stellen jedoch auch eine Herausforderung dar, selbst einmal etwas — zum Beispiel im mitgelieferten TI-Basic — zu programmieren. Der Befehlsvorrat des einfachen TI-Basic wird erheblich erweitert durch das Modul TI Extended Basic, das zusammen mit einem englischen Handbuch für zirka 330 Mark angeboten wird.

Bild 2. Das Modul »TI Extended Basic« wird in den Eingabeschacht an der Frontseite des TI 99/4A gesteckt

Zum Einstieg ein Basic-Lehrgang auf Kassette

Wer mit der Bedienungsanleitung für das einfache TI-Basic nicht zurechtkommt, findet in dem Kurs «Teach Yourself Basic«, der auf Kassette geliefert wird, eine große Hilfe. Die einzelnen Befehle und Funktionen werden in acht Lektionen erklärt, und das eben Erlernte wird in Multiple Choice Aufgaben abgefragt. Mit dem Computer lernen bedeutet jedoch nicht nur Programmieren zu lernen. Mehrere spielerische Lernprogramme sind eigens für Kinder konzipiert. Übungsprogramme für die Grundrechenarten gibt es sowohl für Kinder im Kindergarten — wie auch im Schulalter. Die Kinder können den Schwierigkeitsgrad der Aufgaben selbst bestimmen. Die Programme loben die Übenden sowohl akustisch als auch grafisch, wenn sie eine Aufgabe richtig gelöst haben. War die Antwort falsch, werden die Kinder aufgefordert, es nochmal zu versuchen. Können sie die Aufgabe gar nicht lösen, gibt der Computer selbst die Antwort. Einige der Spiele und Lernprogramme arbeiten mit Sprachausgabe über einen Sprach-Synthesizer. Dies ist ideal für Kinder, die noch nicht lesen können, da der Computer nun mit ihnen »spricht«, und sie ihre Antwort über den Joystick eingeben können. Der Sprach-Synthesizer wird an einen 44-poligen Stecker an der rechten Seite der Konsole aufgesteckt.

Der TI 99/4A bietet jedoch auch Erwachsenen mehr als lediglich Unterhaltung und das Erlernen der Grundkenntnisse der Programmierung. Er kann im Haushalt eingesetzt werden, um Informationen verschiedener Art zu speichern und auszuwerten. So ist es möglich, Einkünfte und Ausgaben zu verfolgen oder ein Archivierungssystem aufzubauen: Eine kleine Datenbank erleichtert den Überblick über ei-

	Großhandel	TV-Geschäft	Bürobedarfsgeschäft
TI 99/4A	563,87	548,—	579,—
Kassettenrecorderkabel	40,62	49,—	49,—
RS232-Schnittstelle	450,87	548,—	466,—
32-KByte-RAM	676,87	998,—	679,—
Floppy-Disk-Steuerkarte	563,87	678,—	594,—
P-Code	789,87	848,—	848,—
Peripheriebox mit Laufwerk	k.A.	2077,—	1779,—
Joysticks	79,—	99,—	89,—
Sprach-Synthesizer	281,37	498,—	296,—

Tabelle 1: Mehrere Einkaufsmöglichkeiten — unterschiedliche Preise (in Mark inklusive Mehrwertsteuer; erhoben bei je einem zufällig ausgewählten Händler der angegebenen Kategorien)

ne Bücher- oder Briefmarkensammlung. Außerdem können die häusliche Inventarliste oder PKW-Rechnungen und Termine wie die nächste TÜV-Vorfahrt auf Kassette gespeichert werden. Das Sichern und Auswerten von Aktienkursen ist eine äußerst interessante Anwendung, die dieser Heimcomputer zu leisten vermag. Doch Vorsicht: Datenhaltung auf Kassette ist eine sehr mühsame Angelegenheit. Man muß sich erst die Position auf dem Band suchen, an der die gewünschten Daten stehen. Dann dauert das Einlesen der Informationen einige Minuten. Gelingt das Laden beim ersten Versuch, hat man schon Glück gehabt. Häufig muß man erst Tonhöhe und Lautstärke kombinieren, ehe die Daten fehlerlos eingelesen werden. Wesentlich einfacher ist es die Daten auf Diskette abzuspeichern; dazu braucht man jedoch ein Disketten-Laufwerk.

Disketten-Laufwerk erleichtert Datenhaltung

Dann muß man allerdings schon etwas investieren; denn um das Diskettenlaufwerk einsetzen zu können braucht man die Erweiterungsbox (Bild 3; Preise: Tabelle 1). Die Erweiterungsbox wird an den 44-poligen Stecker, an den auch der Sprach-Synthesizer gesteckt werden kann, angeschlossen. Sie ist relativ schwer und ihr Ventilator verursacht ziemliche Geräusche. Mit diesem peripheren Erweiterungssystem kann man einfach beginnen und sich nach und nach verschiedene Karten dazu kaufen, die das System um gewünschte Funktionen erweitern. Die Box bietet acht Einschubfächer. In Bild 3 ist die geöffnete Box mit mehreren Karten und dem Anschlußkabel zu sehen. Eine Karte ist nötig, um die Kommunikation zwischen Box und Computer herzustellen. Weitere Karten werden in beliebige Steckplätze gesteckt. Das Problem, welche Karte auf welchen Steckplatz gehört, — wie man es häufig auch bei größeren Computern findet — taucht nicht auf, da die Steckplätze beliebig belegt werden können. Ein Disketten-Laufwerk kann integriert sein, muß aber nicht. Auf jeden Fall ist zum Betrieb eines Laufwerks die Diskettensteuerkarte nötig. Diese verwaltet maximal drei Disketten-Laufwerke. Die 5¼-Zoll-Disketten bieten lediglich eine Kapazität von 89 KByte und werden einseitig, in einfacher Dichte beschrieben. Der Disk Manager, ein vorprogrammiertes Modul, verwaltet Dateien und stellt Diskettendienstprogramme wie beispielsweise »Ausgeben des Inhaltsverzeichnisses« zur Verfügung.

Für Programme, die auch »kommerziell« genutzt werden können, wie beispielsweise Multiplan oder TI Writer, ist neben der Diskettenstation eine Speichererweiterungskarte (32 KByte RAM) erforderlich. Den Anschluß weiterer serieller Geräte wie zum Beispiel eines Druckers erlaubt die RS232-Schnittstellenkarte. Außerdem besitzt diese Karte eine 16-polige Parallel-Anschlußstelle für Matrixdrucker mit Centronics-Port.

Diese Gerätekonfiguration — TI 99/4A, Erweiterungsbox, Diskettenlaufwerk und die erwähnten Karten — gestattet eine für den Hausgebrauch äußerst komfortable Nutzung. Die Kehrseite der Medaille: Mindestens 4000 Mark für Konsole, Box mit Floppy-Disk-Laufwerk, Disk-Controller-Karte, RS232-Schnittstelle und 32-KByte-RAM-Erweiterung. Zusätzliche Kosten verursachen Drucker und Software. Basic ist nicht die einzige Sprache, die der TI 99/4A versteht. Für die Gerätekombination mit Diskette und Speichererweiterung gibt es einen Editor-Assembler, der die Programmierung in TMS-9900-Assembler erlaubt. Mit dieser Gerätekonfiguration läuft auch Logo, eine Computersprache für Kinder, die zum Erlernen der Grundlagen der Programmierung sowie mathematischer und logischer Grundbegriffe entwickelt wurde.

Die p-Code-Karte ermöglicht die Programmiersprache Pascal, unter der UCSD-Pascal-Version 4.0 und TI Pilot, eine Sprache, in der Programme für den Unterricht in der Schule geschrieben werden können.

Die hier erwähnten Erweiterungskarten machen aus dem kleinen TI einen stattlichen Computer. Dafür muß der Benutzer aber ganz schön tief in die Tasche greifen. Einen Preisüberblick vermittelt Tabelle 1. Sie zeigt auch, daß es keine festen Preise gibt. Jeder Händler hat seinen Preis. Wir haben uns bei je einem Großhändler, einem Rundfunk- und Fernsehgeschäft und einem Vertreter des Bürofachhandels nach den Preisen erkundigt — sie sind keineswegs repräsentativ. Tip: Bevor Sie einen Heimcomputer kaufen, erkundigen Sie sich nach den Preisen. Das gilt nicht nur für den TI 99/4A. Beachten Sie auch die Angebote des Versandhandels. Aber denken Sie daran: Ein zuverlässiger Händler, der Ihnen auch mal »auf die Schnelle« ein Kabel, ein seltenes oder neues Softwaremodul oder sonst etwas besorgen kann, ist mehr wert als einmal gesparte 50 Mark. (gu)

99er magazine — TI Magazin

Anregungen und viele Programme für den TI 99/4A findet man im 99er magazine, das monatlich in den USA erscheint. Dort finden sich auch viele Hinweise auf Produkte, die in Deutschland nicht ohne weiteres erhältlich sind.

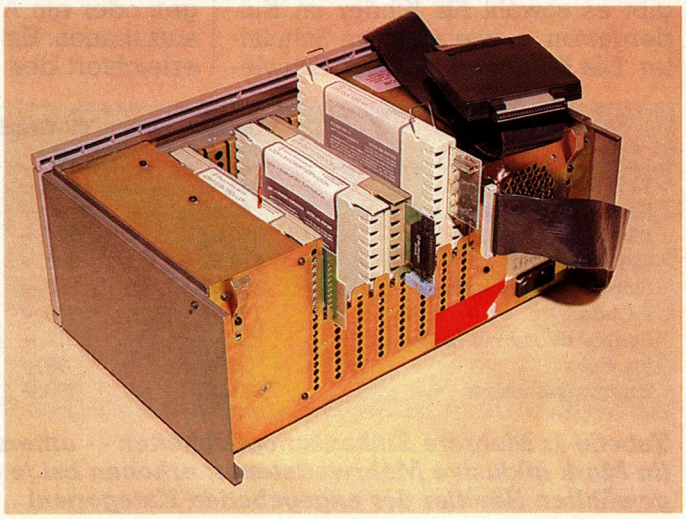
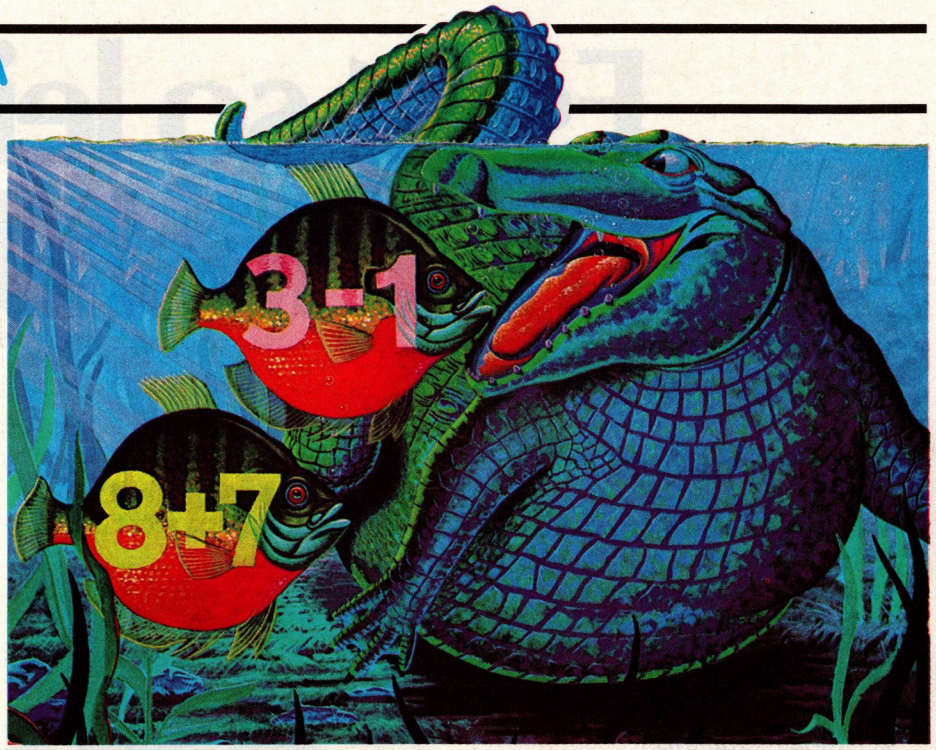


Bild 3. Geöffnete Peripherie-Box mit mehreren Karten, die das System um gewünschte Funktionen erweitern

Lieber und leichter lernen mit dem Computer?



Mit dem Computer spielend lernen — das ist die Idee, die Texas Instruments mit seinen Lernprogrammen für Kinder verfolgt. Wir wollten wissen, ob die Programme »Rechenkünstler«, »Meteoriten-Multiplikation« und »Krokodilsmahlzeit« halten, was sie versprechen.

Diese Lernprogramme werden als einfach zu handhabende Einschubmodule angeboten. Sie sollen elementare arithmetische Fähigkeiten, die durch praktische Übungen vertieft werden können, fördern. Man versucht, dies durch Spiele zur Übung der vier Grundrechenarten, kombiniert mit der Anlernung von Entscheidungsfähigkeiten, zu erreichen. Sie bezwecken also quantitative Verbesserungen von Lernweisen und überzuckern dabei zum Teil trockene arithmetische Übungen mit Spielen. Gleichzeitig führen sie die Kinder in die Handhabung des Computers ein.

Beim Umgang mit solchen Lernprogrammen wird Kindern der Computer als Lehrmaschine präsentiert, die darauf programmiert ist, die Fähigkeiten von Kindern zu trainieren, Übungsaufgaben von angemessenem Schwierigkeitsgrad zu stellen, Feedback zu liefern und Informationen über den Wissensstand der Lernenden zu geben.

Hier »programmiert« der Computer also das Kind — diese Verkürzung verweist auf die grundsätzliche Beschränkung dieser Lernprogramme. Denn folgt man beispielsweise den Überlegungen S. Paperts, der für Kinder die »computerreiche Erziehungsumgebung« der Logo-Programme entwickelt hat, stellt sich die Idee, Computer im Bereich von Lernen und Erziehen zu Drill- und Übungszwecken verschiedener Inhalte einzusetzen, als eine konservative und nicht vorurteilslose Vorstellung dar, in der einfach alte Vermittlungsmethoden mit neuer Technologie gemischt werden. Demgegenüber müsse angestrebt werden, den Computer von Kindern so programmieren zu lassen, daß sie qualitativ denken lernen, wenn nämlich die tatsächlichen Möglichkeiten eines Computers genutzt werden. Denn erst auf diese Weise eingesetzt, ermöglicht der Computer vielleicht eine revolutionäre Wende in der Geschichte des Lernens, weil — vereinfacht gesagt — schon vier- bis sechsjährige Kinder in Zukunft auf diese Weise wissenschaftliches Wissen, wie zum Beispiel große mathematische Ideen, als im persönlichen Wissen verankert verstehen lernen können.

Die fünf ausgewählten Lernprogramme laufen auf dem Rechner TI 99/4A, der mit dem PAL-Modulator PHA 2036, dem AC/AC-Adapter AC 9900 N2, dem Speech-Synthesizer PHP 1500 und den Doppel-Joysticks PHP 1100 ausgestattet sein sollte. Als Monitor diente ein tragbarer Philips-Farbfernseher, der durchaus akzeptable Farb- und Tonqualität lieferte.

Von den ausgewählten Lernprogrammen stehen bis jetzt der »Rechenkünstler« (zirka 50 bis 60 Mark), die »Meteoriten-Multiplikation« (zirka 125 bis 150 Mark) und die »Krokodilsmahlzeit« (zirka 125 Mark) in deutscher Sprache zur Verfügung. Numeration II und Math Games VI erfordern Grundkenntnisse der englischen Sprache, über die nach meinen Erfahrungen die deutschen Kinder in dem für die Lernprogramme angegebenen Alter (überwiegend ab 6 Jahre) noch nicht verfügen. Aus diesem Grund beschränke ich mich auf die Besprechung der drei in deutscher Sprache im Handel erhältlichen Mathematik-Lernprogramme.

Ich habe Kinder (5 bis 7 Jahre) aus einem öffentlichen Kindergarten sowie einige Schüler (8 bis 15 Jahre beziehungsweise 2. bis 9. Klasse) gebeten, die drei Mathematik-Lernprogramme zu testen. Obwohl sie bisher keinerlei Erfahrungen im Umgang mit Computern hatten, zeigten und entwickelten diese Kinder keinerlei Skepsis gegenüber dieser wunderlichen Maschine (warum auch?). Nach wenigen Erklärungen schalteten sie selbst den Fernseher und den TI 99/4A an, legten das Programmmodul ein, wozu allerdings meist die ganze Kraft beider Hände benötigt wurde, und harrten gespannt der kommenden Bilder und Geräusche. Konnten sie zwischen Fernbedienung (Joysticks) und Tastatur wählen, bevorzugten sie die Joy-

sticks, da ihnen die Handhabung der Tastatur doch ein wenig schwerfiel. Hier bringt vielleicht die uns ins Haus stehende Technik der Gemischtmedienspiele Abhilfe; etwa in der Form, daß ein Computer mit Laut- und Sichtzeiten einen Spielvorgang beispielsweise auf einem Brett regelt und Plastikmännchen oder andere, leicht faßbare Tastelemente manipuliert werden.

Nachdem im übrigen selbst ein Eineinhalbjähriger, den ich außer Konkurrenz testete, begeistert gewaltige, bunte Dinge am Monitor zustande brachte, erinnerte ich mich, daß der Feind ja nicht in diesen Jahrgängen zu suchen ist. Laut einer Untersuchung aus Großbritannien ist der »Durchschnitts«-Computer-Käufer 44 Jahre alt, verheiratet, mit 2,2 Kindern und umgerechnet 75 000 Mark per annum gesegnet, ein »gestandenes Mannsbild« also.

Bei der Bewertung der Lernprogramme bediene ich mich gern der Kriterien von Thomas W. Malone, der aus der Analyse der großen Faszination einiger Computerspiele Elemente gewonnen hat, die bei der Entwicklung von Lernprogrammen zu berücksichtigen sind, wenn das Lernen mit dem Computer Spaß machen soll — und das soll es.

Hierzu gehört erstens die Herausforderung als klare Zielsetzung, ohne die kaum ein spielerisches Element in eine Übung zu bringen ist. Sie sollte gleichzeitig für das Kind persönlich bedeutsam sein. Wichtige Momente dieser Herausforderung sind ein veränderlicher Schwierigkeitsgrad, eine Bewertung (Punkte- und/oder Zeitstand) und ein gewisser Glücksspielgehalt.

Zweitens wird eine emotionell ansprechende Phantasiewelt gefordert, und zwar möglichst eine direkt auf die zu erlernende Fähigkeit bezogene.

Drittens und letztens soll die Neugierde der Kinder geweckt und erhalten werden, wozu unbedingt auch ein (vielleicht sogar paradoxes) Überraschungsmoment gehört.

Das Maß aller Dinge über alle didaktischen Forderungen hinaus waren mir natürlich die Kinder selbst mit ihren unübersehbaren Reaktionen, Vorlieben und Ablehnungen.

Schauen wir uns das Rechenkünstlerprogramm näher an: Nach den Angaben von Texas Instruments ist das Rechenkünstlermodul gedacht für Kinder ab 6 Jahre. Das Programmodul soll den Kindern zeigen, daß Rechnen Spaß mache und eine reizvolle Aufgabe sei. Das Modul bietet drei verschiedene Grundprogramm-Möglichkeiten an: »Quick-Quiz«, »Mathe-Quiz« und »Magische Zahl« mit Rechenübungen in den vier Grundrechenarten.

Rechenübungen in den vier Grundrechenarten mit »Quick-Quiz«, »Mathe-Quiz« und »Magische Zahl«

Kein Lernspiel also, sondern Rechenaufgaben, die sich von solchen, die im Heft gemacht werden müssen, durch die Umgebung, in der sie gestellt und gelöst werden, unterscheiden.

Nachdem eine beliebige Taste gedrückt und mit der ersten Programmanweisung »3 for Rechenkünstler« die deutsche Version gewählt ist, entsteht auf dem Bildschirm aus den Requisiten eines Zauberkünstlers ein farbiges Titelbild. Ein Zauberstab läßt das berühmte Kaninchen aus dem Hut des Zauberkünstlers auftauchen, und eine Menge verschiedener Zahlen purzeln mit lustigen Tönen um den Zauberhut herum. Die Neugier der Kinder ist geweckt für das, was nun kommen soll.

Der Rechner fragt nun, welches der drei Grundprogramme das Kind wählen möchte. Da stimmt natürlich die Struktur nicht ganz, denn Kinder zwischen 5 und 7 Jahren können war 2 und 2 zusammenzählen, aber noch nicht lesen. Nun ja, der Papa kann's ja. Und wenn die Speech-Synthesizer — wie alles Elektronische — auch immer billiger werden, können die es noch besser.

Die beim »Quick-Quiz« vom Rechner beliebig ausgewählten Additions- und Subtraktionsaufgaben (zum Beispiel 3+4=? oder 4+7=?) erscheinen auf einer farbigen Leiste in der Mitte des Bildschirms; oberhalb der Rechenaufgaben hüpft das Kaninchen vom Titelbild jeweils ein farbiges Feld weiter, wenn eine Aufgabe gelöst ist. Dazu muß das Kind über die Tastatur das richtige Ergebnis eingeben. Selbst den Kindergartenkindern bereitete diese Anforderung nach einigen Versuchen keine große Schwierigkeit.

Am unteren rechten Rand des Bildschirms werden die Punkte gezählt (bis 100 Punkte). Die richtige Lösung einer Aufgabe wird mit dem Textkommentar »Richtig! Prima!«, der durch Töne untermalt wird, belohnt. Die Wirkung dieses positiven Feedbacks könnte sicher entweder durch ein gesprochenes Lob oder das Wiederauftauchen des vorher eingegebenen Namens des Kindes erhöht werden.

Das Programm läßt für das Kind drei Versuche der richtigen Lösung zu, für die es jeweils 10, 7 oder 4 Punkte erhält. Diese Art der Bewertung war für die Kinder ohne Erläuterungen nicht zu durchschauen.

Löst das Kind eine Aufgabe falsch, wird es durch die Aufforderung »Leider falsch, versuch's noch einmal!« in einer aufmunternden Form zu einem neuen Lösungsversuch ermutigt; die diese Aufforderung begleitenden Töne ermöglichen auch einem Kind, das nicht lesen kann, seinen Fehler zu bemerken. Nach dem dritten falschen Versuch erscheint unter der Rechenaufgabe das richtige Ergebnis sowie der Hinweis »Keine Punkte«. Die endgültige Bewertung der Leistung in diesem Programm wurde als farbige Grafik gestaltet. Nach dem Schema 100% = ausgezeichnet und so weiter wird sie mit entsprechenden Sound-Effekten aus dem Hut des Zauberkünstlers gezogen.

Danach gibt der Rechner an, daß das Kind jetzt mehrere Wahlmöglichkeiten habe; es kann weiterspielen, ein anderes Programm wählen, die bekannte Reihe der Aufgaben wiederholen oder aufhören.

Die Herausforderung beim »Quick-Quiz« besteht in dem einfachen, aber wenig phantasievollen Ziel, seine Fähigkeiten und Kenntnisse zur Lösung einfacher arithmetischer Probleme einzusetzen. Diese Art von Herausforderung ist keine für das Kind persönlich bedeutsame. Insgesamt wird eine die Kinder emotional ansprechende Phantasiewelt nur in Form einzelner dekorativer Elemente, wie dem Titelbild, dem Kaninchen, das dem

Kind anzeigt, wieviele Aufgaben schon gelöst wurden, und der grafisch gestalteten Endbewertung entworfen. Dennoch war insbesondere für die jüngeren Kinder das Kaninchen ein wichtiges grafisches Mittel, an dem sie sich immer wieder orientierten, um den Stand ihrer Bemühungen einschätzen zu können.

Positiv hervorzuheben sind die jeweils mit Toneffekten gestalteten Formen des Feedbacks, negativ die fehlende Möglichkeit, den Schwierigkeitsgrad variieren zu können.

Das fehlende spielerische Moment in diesem Lernprogramm ließ die Neugier und das Interesse der Kinder, die damit arbeiteten, schnell erlahmen; freiwillig wiederholten sie es nicht.

Das »Mathe-Quiz« ist ein etwas komplexeres Programm zur Verbesserung der mathematischen Fähigkeit und der Geschwindigkeit beim Lösen von Rechenaufgaben. Leider verändert sich das schon bekannte Szenario (Gestaltung der Grafik, Toneffekte und Feedback) nicht, die Neugier der Kinder wird auf dieser Ebene nicht erregt — kein Überraschungsmoment nimmt sie erneut gefangen, aber das Kind kann zwischen zwei Möglichkeiten wählen: Entweder kann es mit dem Speicher arbeiten, in den es bis zu zehn Aufgaben zur Beantwortung abspeichern kann (ist allerdings etwas kompliziert) oder der Rechner wählt zehn Aufgaben aus. Dabei können vier verschiedene Schwierigkeitsgrade (Stufe 1 am leichtesten, Stufe 4 am schwersten) sowie die Rechenart (+, —, x, :) gewählt werden. Auch die Form der Aufgabe kann von dem Kind bestimmt werden, entweder die fehlende Zahl oder das fehlende Ergebnis einzusetzen, und es kann auch gegen die Uhr arbeiten.

Obwohl nicht phantasievoller gestaltet, hielt die Herausforderung dieses Programms für die Kinder verschiedener Altersstufen (bis zu 12 Jahren), die damit arbeiteten, viel länger an als beim »Quick-Quiz«. Während des Programms korrigierten die Kinder ihre in einigen Fällen zu anspruchsvoll gewählte Selbsteinschätzung ohne sichtbare Frustration, genossen das Erfolgserlebnis auf einer niedrigeren Stufe und probierten dann noch einmal eine schwierigere Stufe.

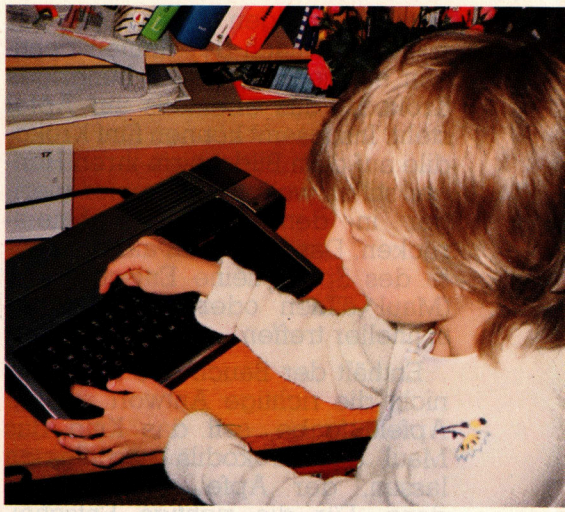
Der selbständige Umgang mit dem Programm erfordert von dem Kind auch die Fähigkeit, schrittweise den Dialog mit dem Rechner zu führen, was neben einer ausgeprägten Lesefähigkeit ebenfalls ein Grundverständnis der Funktionen des Computers voraussetzt. Es ist erstaunlich, wie schnell Kinder nach wenigen Erläuterungen den Rechner handhaben lernen. Sie begreifen die »Computerlogik« viel eher als Erwachsene.

Das Programm »Magische Zahl« variiert die Möglichkeiten des »Mathe-Quiz« nur geringfügig: Das Kind kann für die vier Grundrechenarten jede Zahl zwischen 0 und 9 (eine »magische Zahl«) wählen und wiederum festlegen, ob es gegen die Uhr spielen will, wie die Aufgaben aussehen sollen und welche Rechenart es wählen möchte. Grafik-, Ton- und Feedback-Gestaltung kennen wir schon von den anderen Programmen.

Die Kinder arbeiteten am liebsten mit dem »Mathe-Quiz«. Zwar bietet auch dieses Lernprogramm keine besonders unterhaltsame Lernmöglichkeit, läßt jedoch insbesondere gegenüber dem »Quick-Quiz« den Kindern einen größeren Spielraum für eigenständige Entscheidungen über den Ablauf des Programms. Die Gestaltung der Phantasieumgebung des in seiner Struktur als Drillprogramm entworfenen »Rechenkünstlers« ist mit Einschränkungen nicht besonders gelungen (möglicherweise verstärkt sie im Unbewußten der Kinder noch die falsche Analogie: Rechnen = Zaubern). Das Urteil der Kinder: »Nicht so schlecht, aber eher wie Rechenunterricht ohne Lehrer.«

Für das Rechenkünstlermodul liegt eine ausführliche Dokumentation in deutscher Sprache vor, die einen umfangreichen Anleitungsteil für Kinder »Du und der Rechner« enthält. Dieser Teil ist verständlich formuliert und reichhaltig illustriert. Keines der Kinder, die mit dem Modul arbeiteten, wollte — trotz entsprechender Aufforderung — die Anleitung parallel zur Arbeit am Rechner lesen; das war ihnen viel zu langwierig. Sie wollten gleich mit dem Rechner loslegen, kamen sie nicht weiter, verlangten sie mündliche Erklärungen oder versuchten selbsterfundene Zugriffswege.

Die beiden anderen Programmmodule »Krokodilsmahlzeit« und »Meteoriten-Multiplikation« sind dagegen als Lernspiele konzipiert.



Der Umgang mit der Tastatur fällt den Kindern schwer

Lernen mit »Krokodilsmahlzeit« und »Meteoriten-Multiplikation«

»Krokodilsmahlzeit« (1982, Autoren: J. Chaffin und B. Maxwell, University of Kansas) ist den typischen Spielhallen-Games nachempfunden; es soll durch Herausforderung und Spaß so ganz nebenbei die Additions- und Subtraktionsfähigkeiten mit Zahlen von 0 bis 9 bei Kindern vom Kindergartenalter an verbessern. Es gilt, hungrige Krokodile mit Äpfeln zu füttern, etwas absurd, aber ganz lustig. In den Äpfeln finden sich Additions- und Subtraktionsaufgaben. Auf dem Bildschirm taucht jeweils am linken Rand ein solcher Apfel auf und bewegt sich mit Pfloff-Ploff-Tönen Schritt für Schritt unaufhaltsam auf das Maul des Krokodils zu. Bevor ein solcher roter Aufgaben-Apfel das Maul des grünen Krokodils erreicht, muß das Kind entscheiden, ob die Zahl, die im Bauch des Krokodils zu erkennen ist, das Ergebnis der Rechenaufgabe in dem Apfel darstellt. Ist dies der Fall, muß es (über die Leertaste der Tastatur oder die Aktivierungstaste des Joysticks) schnellstens das Maul des Krokodils öffnen, dem nämlich nur solche Äpfel bekommen. Ein Schmatzton sowie ein Funke am Maul des Krokodils signalisieren dem Kind die richtige Entscheidung und verzeichnen sie als Treffer unten auf dem Bildschirm. Nach den ersten drei richtigen Entscheidungen tauchen die Augen eines zweiten Krokodils auf, nach der

vierten richtigen der Kopf und schließlich nach der fünften richtigen ein ganzes zweites hungriges Krokodil, das ebenfalls auf Äpfel steht. Insgesamt können fünf Krokodile auf dem Bildschirm ihren Hunger stillen wollen, die immer gieriger in den Bildschirm hineinrücken. Das Kind muß also im Laufe des Spiels seine Entscheidung (Maul öffnen oder nicht) immer schneller treffen.

Enthält der Bauch des Krokodils nicht die richtige Antwort auf die Apfelaufgabe, soll das Kind das Maul des Krokodils geschlossen lassen, der Apfel verschwindet dann. Für die richtige Entscheidung in einem solchen Fall wird das Kind ebenfalls mit einem Treffer belohnt. Bei falschen Entscheidungen drückt das Krokodil sein Mißfallen durch einen Brummton aus oder das zweite Krokodil beispielsweise verschwindet bei mehreren falschen Lösungen wieder Stück für Stück. Falsche Entscheidungen werden ebenfalls auf dem Bildschirm angezeigt. Bei einer Fehlentscheidung erhöht sich die Zahl der richtigen Lösungen, die ein zweites Krokodil ins Spiel bringen, von 5 auf 10. Die laufende Anzeige des Punktestands ermöglicht dem Kind, sich schnell über den Spielstand zu informieren.

Es gibt zwei Möglichkeiten für dieses Spiel:
1. Das Spiel beginnt, nachdem eine beliebige Taste gedrückt ist; es dauert zwei Minuten und ist auf einem relativ hohen Schwierigkeitsgrad eingestellt. Die Versuche, die Kinder mit diesem eingestellten Spiel beginnen zu lassen, scheiterten (auch bei den 12jährigen), weil es sie zunächst überforderte.
2. kann über die Anwahl der »Spielwiese« über die AID-Taste vor Spielbeginn von jedem Kind individuell der Schwierigkeitsgrad in bezug auf Zahlenumfang und Geschwindigkeit, Dauer des Spiels, Joysticks J/N eingestellt werden, was sich wiederum als sinnvoll und wichtig erwies.

Das Krokodil mag nur Äpfel mit richtigen Aufgaben

Am Schluß des Spiels gibt der Rechner dem Kind seinen endgültigen Punktestand bekannt. Die grafisch sehr einfallslos gestaltete Punktestandtabelle gibt bei mehreren Runden den jeweils letzten Stand sowie die schlechteste und beste Lösung, die überhaupt erreicht wurden, an.

Das Spiel kann vorzeitig über das Drücken der Taste T beendet und über die AID-Taste wieder neu eingestellt oder über das Drücken einer beliebigen Taste in der von Rechner eingestellten Weise gespielt werden.

Bei diesem Lernspiel ist es offensichtlich gelungen, »Drill« und Spaß in einer für die Zielgruppe (Kindergartenkinder bis zu Schülern in den ersten Schulklassen) unterhaltsamen Weise zu verbinden.

Es gibt eine klare Zielsetzung, den Hunger der Krokodile nach bekömmlichen Äpfeln zu stillen, die nach dem Ausrechnen der Aufgabe mit jeweils einer Entscheidung (Maul öffnen J/N) und einer entsprechenden Handlung auf der Tastatur oder mit dem Joystick erreicht werden kann. Schade, daß die Kinder diese Zielsetzung weder aus den Anweisungen des Rechners, der Spielwiese noch aus der Beobachtung des grafisch ansprechend gestalteten Spielgeschehens auf dem Bildschirm ohne Erklärungen verstehen können. Haben sie jedoch die Zielsetzung verstanden, halten sie den Joystick — alle Kinder spielten lieber mit dem Joystick als mit der Tastatur — ganz fest in beiden Händen, beobachten ganz gefesselt den Apfel, murmeln ihre Überlegungen vor sich hin, und schließlich schreit einer: »Ja, Maul auf, Mensch, drück doch endlich!« Die vielfältigen Toneffekte bereiten den Kindern Vergnügen und motivieren ebenfalls zur Fortsetzung des Spiels.

Weltraumstation mit dem richtigen Lösen von Multiplikationsaufgaben retten

Die nüchterne Punktestandstabelle fanden die Kinder enttäuschend, auch ziemlich schwer zu verstehen; ein bißchen mehr Phantasie und ein paar Töne wären hier sehr schön. Sie vermißten eine abschließende wertende Beurteilung ihrer Spielleistung. Aber das Spiel gefiel ihnen sehr — »ganz anders als Schule!«

Auch die »Meteoriten-Multiplikation« (1982, Autoren J. Chaffin und B. Maxwell, University of Kansas) verspricht Spaß bei der Übung von Multiplikationsaufgaben: Ein Schwarm farbiger Meteoriten bedroht eine stilisierte Weltraumstation, aber eine mit der richtigen Antwort geladene Kanone erledigt auf entsprechendes Kommando den Angreifer. Um die Zielsetzung des Spiel zu erfüllen, sind eine Reihe sensomotorischer Fähigkeiten zusätzlich zu der Fähig-

Kinder lassen sich das Spiel erklären

L.U.D.O.P.
→MICRO←

Realisierte Anwendungen für:

HANDWERKSBETRIEBE

- Auftragskalkulation
  Material
  Zeiterfassung
  Stücklisten
- Fakturation
- Angebotsschreibung

HÄNDLER

- Auftragsverwaltung
- Außendienststeuerung
- Lagerverwaltung
- Warenwirtschaftssysteme

VEREINE

- Mitgliederverwaltung
- Abrechnung
- Meldewesen

FÜR ALLE BRANCHEN

- Finanzbuchhaltung
- Lohn- u. Gehaltsabr.
- Textverarbeitung
- Adreßverwaltung
- Viele Individualprogramme
- Verbindung zum Großrechner

Auf Personal Computer,
NCR, OLYMPIA-PEOPLE +
BOSS, RANK-XEROX D.E.C.,
CP/M
sowie Mehrplatzsystemen
RC-Regne-Centralen
MS-DOS
TURBO-DOS

L.U.D.O.P.
→MICRO←

Reichenbachstr. 16
8000 München 5
Tel. (089) 266129 od. 266145

COUPON

ICH INTERESSIERE MICH

FÜR: .....

FA.: .....

STR.: .....

PLZ/ORT: .....

TEL.: .....

keit, Multiplikationsaufgaben lösen zu können, hier notwendig; das Spiel wendet sich an Kinder ab etwa 8 Jahren.

Um einen Meteoriten zu vernichten, muß das Kind, folgt es zunächst dem vom Rechner eingestellten Spiel, über die Tastatur die richtige Lösung der Aufgabe in die Kanone eintippen, dann die Kanone über die Tasten »S« oder »D« nach rechts oder links bewegen, um sie auf den Meteoriten zu richten und dann über die Leertaste abfeuern. Viel zu kompliziert — selbst für 12- bis 15jährige Kinder — aber über die schon bekannte »Spielwiese« kann man sein Spiel wieder selbst einstellen und auch die Joysticks wählen: Mit dem Hebel des Joysticks kann man die Kanone bewegen und mit dem Aktivierungsknopf abfeuern. Etwas einfacher, aber die Hebel der Joysticks ließen sich nur schwer und langsam bewegen. Das Spiel wird mit den Joysticks auch etwas anders gespielt, da die Antwort für die in den Meteoriten gegebenen Aufgaben in der Raumstation durch den Rechner vorgegeben ist und nicht verändert werden kann. Teilweise ist die Zahl in der Raumstation gleichzeitig das Ergebnis für mehrere Meteoritenaufgaben, so daß das Kind strategisch entscheiden muß, welcher Meteorit die Raumstation im Moment gerade am meisten bedroht.

Die Bewertung während und am Ende des Spiels kennen wir schon — nichts Neues. Falls sich schon während des Spiels herausstellt, daß die Kinder sich überschätzt haben, können sie wiederum über die Taste »T« das Spiel abbrechen und eine andere Einstellung vornehmen.

Fasziniert waren die Kinder schon, die Jungen mehr als die Mädchen, aber alle ein wenig überfordert in der Handhabung: »Wir wollen lieber noch mal das Krokodilspiel machen!«

Aber muß es denn unbedingt wieder dieses Weltraumbrimborium auch bei Lernspielen sein? Auch die Anleihe bei den Kriegscomputerspielen erfreut nicht sehr, wenn man sich sinnvolle Erziehungsziele als Ausgangspunkt für die Konzeption des spielerischen Rahmens wünscht. Die relativ kompliziert zu handhabende Spielumgebung verweist auf ein Problem solcher Lernspiele. Unter Umständen führt ein krasses Mißverhältnis zwischen den Anforderungen an die sensomotorischen Fähigkeiten und an die zu übenden inhaltlichen Fähigkeiten zum Gegenteil des Erstrebten: Die Kinder sind frustriert und resignieren.

Dieses Problem scheint mir in den Math Games VI (Englisch) besser gelöst zu sein. Hier werden außerdem in einem schrittweisen Dialog zwischen Rechner und Kind die Spielbedingungen in einer verständlichen Weise festgelegt.

Das hungrige Krokodil machte den Kindern am meisten Spaß

Für die »Krokodilsmahlzeit« (etwas verwirrend, daß auf dem Titelbild ein hungriges Krokodil nach Fischen schnappt, logischer zwar, aber nicht dem Spiel entsprechend) und die »Meteoriten-Multiplikation« liegen ausführliche Dokumentationen nur in englischer Sprach vor.

Fazit: Die drei in deutscher Sprache erhältlichen Mathematiklernprogramme aus meiner Auswahlkollektion hielten nicht immer, was sie versprachen. Nach den Beurteilungen der Kinder und den entworfenen Ansprüchen an Lernprogramme ist die »Krokodilsmahlzeit« am besten gelungen — dennoch: Die Entwicklung von Computerlernprogrammen für Kinder sollte strukturell eher an der Schildkröte von Logo als am hungrigen Krokodil orientiert werden. Bleibt noch die Gretchenfrage, was denn die Kinder eigentlich lernen mit den Lernprogrammen. Sie erlernen den ersten Umgang mit dem Computer, also einem wesentlichen Bestandteil ihrer zukünftigen Welt. Ein bestimmter Trainingseffekt im Umgang mit den vier Grundrechenarten ist ebenfalls mit Sicherheit festzustellen, und es steht zu hoffen, daß sich dieser auch beim Rechnen ohne Krokodile durchsetzt. Ganz deutlich macht sich auch das Ausbleiben von Aggressionen bemerkbar, die sich bei Kindern gegenüber lehrbeflissenen Eltern und Lehrern einzustellen pflegen, die ihre Kinder zu drillen versuchen — das kann der Computer offensichtlich besser.

(M. Eppenstein-Baukhage)

Sprache für Eltern und Kinder

Texas Instruments wirbt schon seit einem guten Jahr mit der Programmiersprache Logo für den 99/4, und Leser des amerikanischen 99er Magazine wissen um die Verfügbarkeit der US-Version — in der Bundesrepublik kaufen konnten die Fans sie bislang noch nicht. Während die eingedeutschte Version dem Vernehmen nach vor der Auslieferung steht, ist in den Staaten bereits das verbesserte TI Logo II angekündigt. Hier ein Zwischenbericht.

Niedere Schwelle, ohne aber gleich an die Decke anzustoßen«, so charakterisiert Professor Seymour Papert vom Massachusetts Institute of Technology sein Logo. Und in der Tat: Während die Insider der Computer-Kultur Logo gern als »Stützräder« fürs Programmieren-Lernen bezeichnen, birgt diese Sprache soviel Möglichkeiten, daß sie — nachdem die Heimcomputer über immer mehr Speicherkapazität verfügen — in der Popularität das ehrwürdige Basic ablösen kann.

Texas Instruments förderte schon 1970 die Gruppe am MIT (Massachusetts Institute of Technologig). Aber es dauerte dann noch über 10 Jahre, bis die TI-Logo-Steckeinheit für den 99/4 in Amerika herauskam. Der Grund war, daß die ursprüngliche Version zu langsam lief und dann doch »von Hand« in die Maschinensprache des 99/4 übertragen werden mußte. Kaum hatte sich die MIT-Gruppe aufgelöst, gab es dann gleich zwei Versionen auf Diskette für den Apple II. Nach neuesten Nachrichten sind sogar Logo-Versionen in Arbeit, die auf allen CP/M-Rechnern laufen können. Auch Thomson will für seinen TO7 ab Ende dieses Jahres ein Logo anbieten.

Dadurch, daß Logo extrem benutzerfreundlich angelegt ist, ergibt sich ein enormer Speicherplatzbedarf. TI-Logo benötigt daher den 32-KByte-Zusatzspeicher und damit die etwas aufwendige Erweiterungseinheit mit dem Lüfter und ihrer Nabelschnur zur Konsole mit der Tastatur. Trotzdem reichte der Speicherplatz nicht aus, um auch Kommazahlen mit hereinzunehmen: TI Logo rechnet nur mit ganzen Zahlen. Dafür hat TI Logo eine trickreichere Grafik mit Hilfe der sogenannten »sprites«, zu deutsch Kobolde, die das Schirmbild im Gegensatz zur Apple-Version enorm beleben können (siehe unten).

TI Logo wird in einem dicken Ringbuch, Format etwa A4, in dem das Benutzerhandbuch von 83 Seiten und die Steckeinheit untergebracht sind, angeboten. Dazu gibt es eine Bandkassette und eine Diskette mit denselben Beispielprogrammen nebst einer 19seitigen Anleitung »TI Logo Sampler«. Zwei weitere Kassetten und eine weitere Diskette finden in diesem Ordner noch Platz. Das Layout des englischen Textes ist übersichtlich, eine Schnellübersicht zeigt die Unterschiede bei den Befehlen zwischen dem 99/4 und dem 99/4A, und in einem Anhang werden die Logo-Vokabeln in alphabetischer Reihenfolge erklärt. Allerdings wurden ein paar vergessen (siehe nebenstehenden Kasten). Der Listenpreis in der USA beträgt 130 Dollar, bei den US-Discount-Versendern ist die Erstversion schon per Kreditkarte für 75 Dollar zu haben.

Bild 1. Kinder üben mit einem Standardprogramm des TI Logo

Herausforderung für Eltern

Während die städtischen Schulämter noch kreativ überlegen, wie sie die Anschaffung eines Heimrechners pro Schule abwenden können, machen sich vorausblickende Eltern Gedanken, wie sie ihre Kinder dennoch für deren so andersgestaltige Zukunft computerfest machen können. Die Logo-Versionen für Heimrechner bieten derzeit den besten Anfang zum gemeinsamen Programmierenlernen. Noch einfacher dürfte das in den nächsten Jahren nicht zu haben sein. Welche Eltern können sich solchen Sprung ins kalte Wasser zutrauen? Kann man Bekannten ohne jede Computer-Erfahrungen zur Anschaffung von Logo raten, ohne das Handbuch von den Frustrierten beim nächsten Besuch an den Kopf geworfen zu bekommen?

Nach Meinung der Amerikaner ist die Hands-on-Erfahrung (Erfahrungen, die man im praktischen Umgang mit dem Gerät macht) der

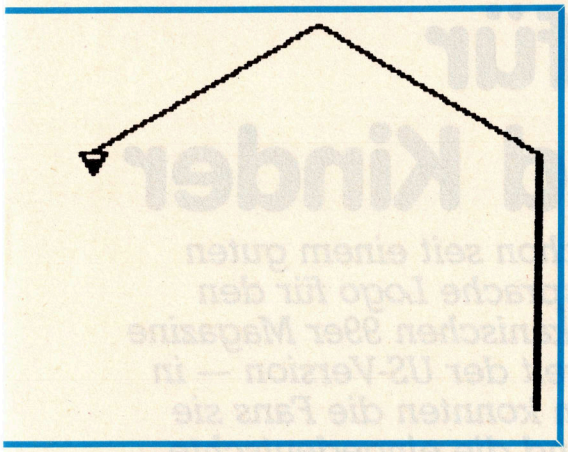


Bild 2. Innenwinkel mit Außenwinkel verwechselt — das war der mißlungene Versuch ein gleichseitiges Dreieck zu zeichnen

beste Lehrmeister — auch für die Eltern. Das ist ja der Hauptvorzug der Lernmaschine Computer schlechthin. Und in der Tat: Setzen Sie Ihr Kind vor den Rechner, und seine Fragen werden Sie so ins Schwitzen bringen, daß Ihr eigenes Lerntempo sich zu beachtlichen Höhen steigert — wenn Sie bereit sind, unter Erfolgsdruck gelassen im Handbuch zu recherchieren. Ein Familienmitglied, das wenigstens minimale Erfolgserlebnisse in der Schulmathematik hat oder hatte, ist da schon von einigem Vorteil.

Andererseits ist der Logo-Mikrokosmos wirklich kindgerecht. Amerikanische Kindergarten-Kinder versetzen sich tanzend in die Bewegungen des grafischen Cursor-Tiers, der Schildkröte, und übertragen nachher das Erlebte auf den Bildschirm. Bild 1 zeigt Kinder, die gerade mit einem Standardprogramm des TI Logo üben.

Die neue Turtle-Geometrie

Aber selbst technisch vorgebildete Väter sollten jetzt nicht überheblich werden: Ein Bekannter, Professor der Physik, wollte ein gleichseitiges Dreieck in Logo zeichnen. »Nichts einfacher als das«, erklärte er den Kindern. »Also, die Winkelsumme im Dreieck ist 180 Grad, geteilt durch drei gibt 60 Grad. Dann tippe ich:

```
TO DREIECK
  FORWARD 80
  LEFT 60
  FORWARD 80
  LEFT 60
  FORWARD 80
  LEFT 60
END
```

Als er nun nach Eingabe dieses Programms in den Arbeitsspeicher (mit den Tasten FCTN-9) zur Ausführung schritt, bei Logo mit dem kurzen Befehl DREIECK, und ENTER drückte, sah das Ergebnis wie in Bild 2 aus:

Der Herr Professor hatte sich gründlich vertan. Ihm war nicht aufgegangen, daß die Drehungen der Schildkröte den sogenannten Außenwinkeln beim Dreieck entsprechen, nicht den Innenwinkeln. Das richtige Programm (in Logo heißt das »Procedure«, aber im Deutschen denkt jeder gleich an »lästige Prozedur«, deswegen sollte man diese Bezeichnung vermeiden) sieht, gleich unter Verwendung des nützlichen REPEAT-Befehls, nämlich so aus:

```
TO DREIECK
  CLEARSCREEN
  REPEAT 3 [FD 80 RT 120]
  HIDETURTLE
END
```

Wobei hier gleich die fingerschonenden Abkürzungen für die entsprechenden Befehle benutzt und mit HIDETURTLE, abgekürzt HT, auch noch der Turtlecursor zum Verschwinden gebracht wurden. Das Ergebnis ist ein gleichseitiges Dreieck mit 80 Schildkrötenschritten Seitenlänge (Bild 3).

Es ist von vornherein unmöglich, die grafischen Fähigkeiten von Logo auch nur zu streifen, doch verdient noch der Vorteil der englischen Logo-Vokabeln die Aufmerksamkeit des Lesers:

— Die Kinder lernen spielend Englisch und der Englischlehrer ist dankbar, daß es endlich etwas gibt, wo die Kinder ihre Englischkenntnisse einsetzen können.

— Es ergibt eine hervorragende Unterscheidungsmöglichkeit zwischen den unantastbaren Logo-Vokabeln (mit der unschönen Bezeichnung »primitives«) im Englischen und den selbst zu schöpfenden Programmnamen, wenn man letztere deutsch formuliert — wie in obigen Beispielen.

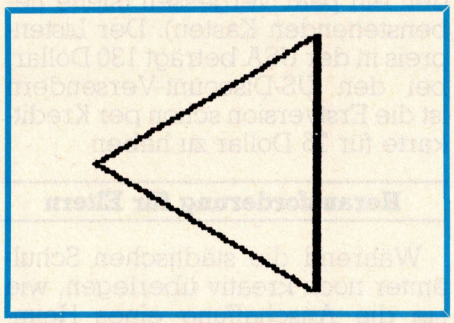


Bild 3. Gleichseitiges Dreieck mit 80 »Schildkrötenschnitten«

Logo auf Deutsch?

Doch halt, wie steht es mit den uns Deutschen so lieben Umlauten und dem scharfen ß aus, fragen jetzt die Eltern mit sorgenvollem Blick in die Diktathefte. Überraschenderweise bietet TI Logo hier die bislang sauberste Lösung an für ein Problem, das bei allen anderen Logo-Versionen wegen der deutschen DIN-Tastaturnorm unlösbar sein dürfte.

Zunächst einmal, die Sorgen sind berechtigt: Nach ein paar Tagen Heimrechner-Gebrauch schrieb unsere Jüngste im Diktat: Tuer, Grossvater, naemlich, Kroete. Die hilfsweise Schreibmaschine der Umlaute prägt sich unglaublich schnell ein! Abhilfe bringen bei TI Logo die sogenannten Tiles, wörtlich Kacheln oder besser Mosaiken. Es handelt sich um 256 Grafikplätze aus 8 x 8 Punkten, von denen 64 bereits mit den Buchstaben und Zeichen der Tastatur belegt sind. Glücklicherweise haben die Tasten für @, #, ≙, und & in Logo keine Bedeutung und können bei TI mit den Umlauten und ß belegt werden; das geht so:

Man schaut in der Buchstabentabelle die ASCII-Kenn-Nummer des Zeichens nach, also etwa 94 für ≙, und geht dann mit MAKECHAR 94 in den »Buchstaben-Entwerfer«. Mit den Pfeiltasten kann man nun ein Ü entwerfen, das mit den Tasten FCTN-9 übernommen wird, (siehe Bild 4).

Im TI Logo gibt es nur Großbuchstaben (TI Basic hat auch Kleinschrift; in Wirklichkeit jedoch nur verkleinerte Buchstaben). Trotzdem fand ich es sinnvoll, für die Kinder ein ß einzuführen, damit ihre Orthographie nicht durcheinander kommt. Die wenigen Fälle, wo bei Großschrift dann noch zwei S stehen müssen, haben sie in der Schule anstandslos bewältigt.

Nachdem die vier Tasten mit den deutschen Zeichen belegt sind, empfiehlt es sich, diese Mosaiken auf Diskette (oder Band) etwa unter dem Namen DEUTSCH zu retten, damit sie bei der nächsten Sitzung nur eingespielt zu werden brauchen.

Eigenartigerweise hat die bevorstehende deutsche TI Logo-Version hiervon keinen Gebrauch gemacht ►

— das Dreieckprogramm sieht so in ihr aus:

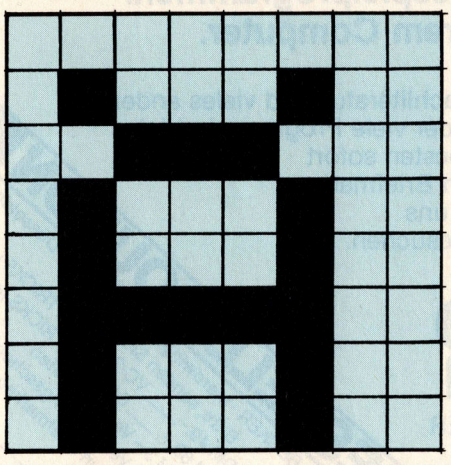
```
PR DREIECK
LOESCHEBILD
WIEDERHOLE 3 [V 80 R 120]
VERSTECKIGEL
ENDE
```

Durch Wechseln des Cursor-Tiers zum Igel hat man die lustigen amerikanischen Illustrationen mit der Schilkröte zu Makulatur werden lassen.

Angesichts des Aufschwungs der internationalen Logo-Bewegung mit User-Clubs und Programm-Austausch ist natürlich ein abweichender Satz von Logo-Vokabeln hinderlich. Wenn weiterhin Logo in Zukunft dem Basic vergleichbare Verbreitung finden soll, wäre das Nachlernen der englischen Vokabeln unausbleiblich. Nach meinen Beobachtungen können sich schon Achtjährige mit den englischen Logo-Vokabeln anfreunden.

Bei den Logo-Versionen für den Apple können keine anderen Tastenbelegungen kreiert werden. Selbst im neuen Apple IIe wechseln getreu DIN das Ä und Ü mit den eckigen Klammern ab. Man kann dann zwar Ä und Ü schreiben, aber Logo erkennt diese als eckige Klammern und schickt Fehlermeldungen. Auf dem Drucker, der meist nur intern umgestellt werden kann (was unterbleibt), sorgen Ä und Ü statt der eckigen Klammern für unübersichtliche Programm-Auflistungen.

Übrigens: Aus dem TI Logo können gegenwärtig nur glückliche Besitzer des nicht mehr lieferbaren Thermodruckers Programmlisten erhalten. Das Ausdrucken der Grafik ist nicht möglich. Dem Vernehmen nach bemüht sich TI Deutschland um einen preiswerten Nachfolger für den Thermodrucker.



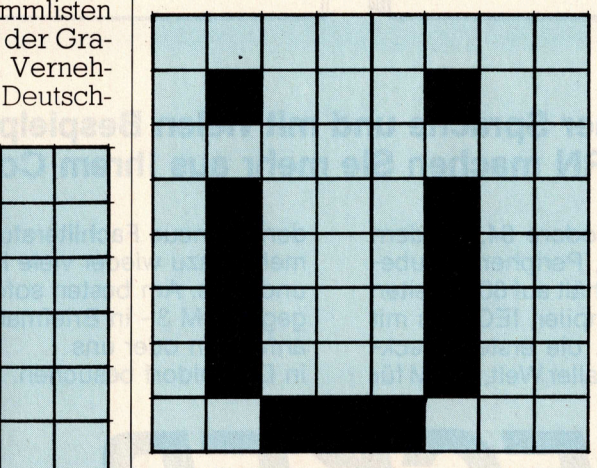


Bild 4. Selbstentworfene Umlaute, mit denen die Tasten @, #, und ^ in TI Logo belegt werden können

Tanz der Kobolde

Über die fehlende Grafik-Hardcopy vermögen die Sprites, zu deutsch Kobolde, hinwegzutrösten. TI Logo hat 32 solcher unsichtbaren dienstbaren Geister, denen man noch ihre grafische Ausformung aus 16 x 16 Bildpunkten, eine von 16 Farben sowie Bewegungsrichtung und -geschwindigkeit erteilen kann. Fünf Formen sind ins Programm eingebaut: Flugzeug, Rakete, Ball, Lastwagen und Kiste. Die erste Demonstration, wenn alle diese Objekte in verschiedenen Farben kreuz und quer über den Bildschirm fahren, fasziniert jeden! Nebenher kann übrigens noch ein Programm mit unverminderter Geschwindigkeit laufen. Beim Übereinanderfahren der Objekte gibt 's Prioritäten, wenn etwa der gelbe Ball als Sonne hinter dem Flugzeug durchwandert, so daß wirkliche »lebende Bilder« mit Tiefenwirkung entstehen. Die Zuweisung an die Kobolde geschieht wie folgt:

```
TELL SPRITE 1
CARRY :ROCKET
SETCOLOR :GREEN
SETSPEED 60
SETHEADING 90
```

Worauf also eine grüne Rakete mit der Geschwindigkeit 60 ständig von unten nach oben fliegt. Weitere grafische Formen kann man mit MAKESHAPE in einem »Grafik-Entwerfer« gestalten und als SHAPE auf Diskette oder Band retten. Hier ergeben sich ungeahnte Möglichkeiten für grafische Spiele, wozu das Handbuch noch mehr Anregungen bringen sollte.

Die deutsche Version textet bei den Sprite-Kobolden etwas an der Benutzerpsyche vorbei:

```
AN FEE 1
TRAGE :RAKETE
SETZFARBEN :GRUEN
SETZTEMPO 60
SETZKURS 90
```

Programmierende Kinder wollen nicht unterstellt bekommen, daß sie noch an die Märchenfee — beziehungsweise gleich 32 davon — glauben, und auch die Erwachsenen kommen sich dadurch leicht veralbert vor. Probleme ergaben sich sogar für die Schreiber des vorläufigen deutschen Manuals, nämlich, ob man den Befehl FEENUMMER mit einem oder zwei N schreibt.

Logo — nun auch in deutsch

Auf der US-Computer-Show 1983 in Köln wurde Logo erstmals in deutsch vorgestellt. Vater des deutschen Logo-Programms — das englische von Abelson ist in Schulen in Nordrhein-Westfalen bereits im Einsatz — ist Professor Löthe von der Pädagogischen Hochschule in Stuttgart. Die deutsche Version läuft auf dem Apple II+, Apple IIe und kompatiblen Computern. Sie wird vom IWT-Verlag in Vaterstetten bei München für 598,— Mark (Diskette, Buch, Dokumentation) angeboten.

Mit Listen und Tücken

Während die Kinder bei dem Umgang mit der Schildkrötengrafik schon mal ermüden, zumal sie sie nicht ausdrucken können, erwies sich der Umgang mit Texten als der eigentliche Magnet von Logo. Logo verarbeitet Text- und Zeichenfolgen, die hier als Words, also Wörter, bezeichnet werden. Entsprechend können aber auch noch sogenannte Listen abgearbeitet werden, also eine Folge von Zeichen mit Leertasten dazwischen und von eckigen Klammern eingeschlossen. Damit läßt sich nun alles anstellen, was in Basic mit den undurchsichtigeren Stringvariablen zu machen ist. Eine Liste von drei Vornamen kann man etwa mit VN bezeichnen:

MAKE "VN [ANNE JULIA RENATE]

und dann etwa befehlen:

PRINT BUTLAST :VN

also alle bis auf den Letzten und der Schirm meldet:

ANNE JULIA

Mit SENTENCE kann man Listen aneinanderreihen. Vorher definiert man sich schnell noch eine andere Liste:

MAKE "NN [LESSING]

jetzt folgt der Befehl

PRINT SENTENCE FIRST :VN :NN

und der Schirm zeigt

ANNE LESSING

Der SENTENCE-Befehl hat also FIRST :VN mit :NN verknüpft. Variable sind in Logo mit vorausgehendem Doppelpunkt gekennzeichnet, es sei denn, sie werden soeben definiert. Hier eröffnen sich nun Möglichkeiten, selbst Büffelprogramme für Kopfrechnen und Vokabel-Lernen zu schreiben, wofür dem Handbuch nicht allzuviel Anregungen zu entnehmen sind. Meine bisherigen Erfahrungen zeigen allerdings, daß da zunächst die Eltern ran müssen; die Kinder konsumieren die Büffelprogramme zwar gerne, aber sie wollen sich zunächst keine eigenen ausdenken.

Ein Geburtstagsquiz ist als Beispiel 1 aufgelistet und in Bild 5 zu sehen. TI Logo II wird dann auch das Abspielen eines Geburtstagsliedes ermöglichen, was auch die deutsche TI-Logo-Version bieten wird. In der Erstversion kann nur mit BEEP ein Dauerton eingeschaltet werden, der mit NOBEEP wieder abzustellen ist.

Es gibt aber auch Tücken bei den Listen in TI-Logo. Während die Apple-Versionen negative Zahlen ohne weiteres in einer Liste abarbeiten, passiert hier folgendes: Für

PRINT LAST [10 - 1]

meldet der Bildschirm

1

Auch, wenn Minuszahlen in runde Klammern gesetzt werden (was das Handbuch verschweigt), ist die Welt noch nicht in Ordnung:

PRINT LAST [10 - (1)]

)

Erst auf Umwegen kommt die Sache ins Lot:

MAKE "LISTE [SENTENCE 10 (-1)]
PRINT LAST :LISTE
-1

Es gibt noch weitere Unterschiede zum Apple-Logo, etwa, daß Zif-

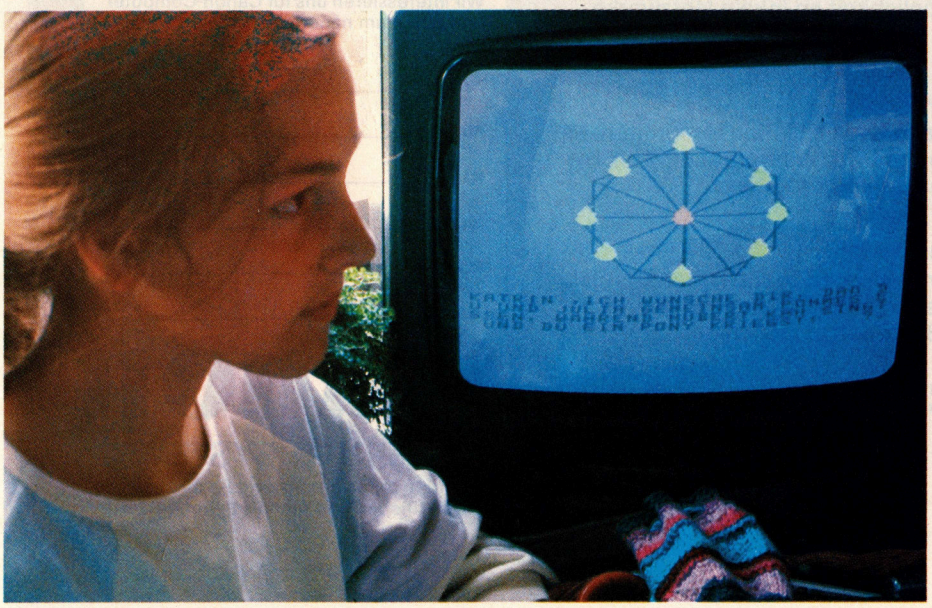


Bild 5. Ein Geburtstagsquiz mit Logo

Im Handbuch vergessene Logo-Vokabeln

In Logo heißen Variable »Namen« und ihre Werte »Sachen«, weil ja als Werte nun außer Zahlen auch Zeichenfolgen, in Logo »Wörter«, und Listen in Frage kommen. Für die Beispiele muß erst noch vereinbart werden:

MAKE "VN [ANNE JULIA RENATE] } Liste
MAKE "MN "ROBERT } Wort
MAKE "JR 1983 } Zahl

Name	Sache
FPUT	**setzt eine Sache in eine Liste als Erstelement. Beispiel: (anschließend an dasjenige von oben)** **PRINT FPUT "AIDA :VN Ergebnis: AIDA ANNE JULIA RENATE**
LPUT	**setzt eine Sache in eine Liste als neues Letztelement. Beispiel:** **PRINT LPUT :MN :VN Ergebnis: ANNE JULIA RENATE ROBERT**
NUMBER?	**gibt TRUE aus, wenn eine Zahl eingegeben wird. Beispiel:** **IF NUMBER? :JR BEEP Ergebnis: es piept** **IF NUMBER? :VN BEEP Ergebnis: es piept nicht** **IF NUMBER? :MN BEEP Ergebnis: es piept nicht**
WORD?	**gibt TRUE aus, wenn ein Wort eingegeben wird. Beispiel:** **IF WORD? :JR BEEP Ergebnis: es piept** **IF WORD? :VN BEEP Ergebnis: es piept nicht** **IF WORD? :MN BEEP Ergebnis: es piept**
THING?	**gibt TRUE aus, wenn die eingegebene Variable gleich einer anderen Variablen ist, die schon einen Wert hat. Beispiel:** **IF THING? :MN BEEP Ergebnis: piept nicht** **MAKE "MA "MN** **IF THING? :MA BEEP Ergebnis: es piept, ebenso, wenn der Wert eine Zahl oder Liste ist.**
DEBUG	**schaltet auf Programm-Prüfung; Schirm meldet ON. Dann Programm laufen lassen: Wo eine Programm-Laus angetroffen wird, hält der Rechner an. Abschalten wieder mit DEBUG; Schirm meldet OFF**

```
TO POLY N L
PD
REPEAT :N [FD :L RT 360/ :N]
END

TO KUCHEN R N L
TELL BG
SC 13
TELL TURTLE
SY 20
REPEAT :R[POLY :N :L RT 360/
:R]
TELL SPRITE 1
CARRY 6
SC 9
HOME
FD 30
SX -9

TELL SPRITE 2
CARRY 6
SC 10
HOME
BK 20
SX -9

TELL SPRITE 3
CARRY 6
SC 10
HOME
SX 35

TELL SPRITE 4
CARRY 6
SC 10
HOME
SX -52

TELL SPRITE 5
CARRY 6
SC 10
HOME
FD 30
SC 45

TELL SPRITE 6
CARRY 6
SC 10
HOME
FD 30
SX -63

TELL SPRITE 7
CARRY 6
SC 10
HOME
FD 60
SX 35

TELL SPRITE 8
CARRY 6
SC 10
HOME
FD 90
SX -9

TELL SPRITE 9
CARRY 6
SC 10
HOME
FD 70
SX -45
END
```

```
TO FUA
CS
TELL :ALL
SC :CLEAR
NOTURTLE
TELL BG
SC 7
TYPE [WAS IßT DU AM LIEBSTEN?]
PC 32
MAKE "A READLINE
PRINT [ ]
TYPE [WER IST DEIN LIEBLINGS
SÄNGER?]
PC 32
MAKE "B READLINE
PRINT [ ]
TYPE [WEN MAGST DU GERN?]
PC 32
MAKE "C READLINE
PRINT [ ]
TYPE [WAS WÜNSCHST DU DIR ZUM
GEBURTSTAG?]
PC 32
MAKE "D READLINE
PRINT [ ]
TYPE [WIE HEIßT DU?]
PC 32
MAKE "E READLINE
TELL TURTLE
TELL BG
SC 13
KUCHEN 12 3 60
TYPE :E
PC 32
TYPE [,ICH WÜNSCHE DIR, DAß DU]
PC 32
TYPE [UND]
PC 32
TYPE :C
PC 32
TYPE [GEMEINSAM]
PC 32
TYPE :A
PC 32
TYPE [ESSEN, WÄHREND]
PC 32
TYPE  :D
PC 32
TYPE [SINGT UND DU]
PC 32
TYPE [KRIEGST!]
PC 32
PC 32
PC 32
GLÜCK; nur für TI Logo II
END

PR GLÜCK
SETTEMPO 240
LEGATO
SETVOICE 0
SETVOICE 1
MAJOR
MUSIC    [4 7 6 5 6 7 4 4 5 4 3
          2 4 2 3 4 5 4 2 4 3 2
          1 1 0] [2 2 2 1 1 2
          2 2 2 2 2 2 2 2 2 2
          2 2 2 2 1 1 2 2 3]
         PLAYMUSIC
END
```

Beispiel 1: Das Oberprogramm ist FUA. Von ihm werden dann die Unterprogramme POLY, KUCHEN und GLÜCK aufgerufen. Die Kerzenflammen im Kuchen werden per MAKESHAPE von Hand entworfen. Formen und Programme werden zweckmäßig zusammen unter einem Dateititel FUA abgespeichert, so daß sie auch gemeinsam eingelesen werden können.

fern bei TI-Logo nicht als Wörter gelten, sofern sie als Wert einer Variablen vorkommen, aber dies wird voraussichtlich in der Version II ausgeräumt sein.

Preiswerter Einstieg

TI Logo ist immer noch der preiswerteste Weg, um zu einem komfortablen Logo mit Listen-Verarbeitung zu kommen. Gegenüber den Apple-Versionen fehlen gegenwärtig Kommazahlen, Grafik-Ausdruck und Musik-Komposition. Letzteres wird die eingedeutschte Version enthalten, die für Kinder unter acht sicher empfehlenswert ist. Ansonsten sollte man auf das amerikanische TI Logo II warten, das größere Formen (32 x 32) für die Sprites und die Ausgabe über die serielle Schnittstellen-Karte RS232, also auf fast jeden Drucker, bieten soll.

(Renate Lessing)

Logo aus USA beziehen

Leser des 99'er Magazine kennen bereits die Anzeigen der US-Discountversender. Der Bezug ist tatsächlich problemlos, besonders mittels der Eurocard, die dem MasterCard-System angeschlossen ist, da der Versender die Portokosten selbst abziehen kann. Sonst muß das Porto im voraus aufgeschlagen werden, wenn man bei der Bank einen Dollarscheck ausstellen läßt (sechs Mark Gebühr). Denn in den USA gilt Vorauskasse. TI Logo I und auch TI Logo II kosten zum Beispiel je 75 Dollar.
Sie schreiben: Please send via air-parcel post (insured)
one TI Logo II at $75
Charge it to my MasterCard No. ????????????? expiring 04-84
Card-Holder: Ihr Name

Die Luftpostgebühr wird bei 10 Dollar liegen. Wenn Sie keine Eurocard oder Visacard haben, fügen Sie noch hinzu:
Enclosed find my cheque on $85 das sind rund 220 Mark. Der Zoll verzollt Sendungen unter 330 Mark mit einem Pauschalsatz von 15 Prozent für Zoll- und Einfuhrumsatzsteuer, also rund 30 Mark.

Basic oder Pascal für den TI 99/4A?

Der TI 99/4A versteht sowohl Basic als auch Pascal. Während insbesondere das TI Extended Basic komfortabel und preiswert ist, enttäuscht Pascal, das hohe Ausgaben für Peripherie nötig macht und dann seinem Preis nicht gerecht wird.

Basic läuft auf dem TI 99/4A bereits in seiner einfachsten Ausstattung, die aus einem Fernseher und dem 99/4A mit PAL-Modulator und Netz-Adapter besteht. Kein Laden des Interpreters ist nötig, da Basic im ROM-Bereich des TI 99/4A enthalten ist. Das Betriebssystem stellt die Befehle BYE, OLD, LIST, RES (equence), NEW, NUM, SAVE, EDIT, DELETE sowie die zur Fehlersuche nützlichen Kommandos TRACE, UNTRACE, BREAK und CONT zur Verfügung. Der Anweisungsvorrat des TI Basic hat den auch in der Grundausstattung anderer Rechner üblichen Minimalumfang. Die beliebten Unterprogramme PEEK und POKE sind ihm jedoch unbekannt. Dafür bietet TI Basic aber die Unterprogramme G CHAR, COLOR, CLEAR und CHAR, die die grafischen Fähigkeiten des Computers voll zur Geltung bringen. Wie wirkungsvoll gerade das Unterprogramm CHAR eingesetzt werden kann, zeigt der Auszug aus dem Buch TI Extended Basic (siehe rechts), in dem ein tanzendes Männchen programmiert wird.

Das erweiterte Basic wird auf einem Programm-Modul zusammen mit einem englischen Handbuch für zirka 330 Mark angeboten. Dieses Modul stellt zusätzliche Anweisungen und Unterprogramme zur Verfügung und erlaubt damit, die Fähigkeiten des Heimcomputers TI 99/4A voll zu nutzen. In Bild 1 sind alle Anweisungen, Unterprogramme und Funktionen aufgelistet, die TI Extended Basic kennt. Außerdem dürfen beim erweiterten Basic mehrere Anweisungen in einer Zeile stehen. Als zusätzliche Vergleichsoperatoren stehen AND, OR, XOR und NOT zur Verfügung, die in der IF-THEN-ELSE-Anweisung unmittelbar ausgeführt werden. Mehrfachzuordnungen der Form X,Y,Z = 1 sind ebenfalls erlaubt.

Ausgerüstet mit einem guten Basic-Buch, wie dem TI Extended Basic von Texas Instruments, dürfte es selbst einem Anfänger keine größeren Probleme bereiten, sich im TI Basic zurechtzufinden. Das Basic-System arbeitet zuverlässig, ist benutzerfreundlich und zudem preiswert.

Pascal hingegen kann dieses Lob nicht zuteil werden. Um überhaupt mit der UCSD-Pascal-Version IV.0 arbeiten zu können, benötigt man neben der Konsole mindestens ein Diskettenlaufwerk, den Floppy-Disk-Controller, die 32-KByte-Speichererweiterung, die P-Code-Karte und schließlich die Peripherie-Box, in die alle Erweiterungskarten gesteckt und das Diskettenlaufwerk integriert werden können. Die Pascal-Software wird auf 4 Disketten (Editor, Filer, Compiler, Assembler, Linker und Utilities) geliefert. Diese Gerätekonfiguration mit Software kostet zirka 5000 Mark. Um jedoch sinnvoll in Pascal programmieren zu können, braucht man ein zweites Diskettenlaufwerk und einen Drucker. Wer sein System derart ausrüstet, muß dafür mindestens 7000 Mark (je nach Drucker) bezahlen. Dann hat man zwar einen Heimcomputer, der Pascal versteht, aber nur 48 KByte RAM zur Verfügung, lediglich 89 KByte Diskettenkapazität und eine Bildschirmausgabe von 40 Zeichen pro Zeile und 24 Zeilen. Alle Mängel, die in diesem Pascal-System auftreten, lassen sich auf die aufgeführten Engpässe zurückführen.

Anwendern ohne Computerkenntnisse dürfte es schwer fallen, mit diesem System zurechtzukommen. Die englischsprachigen Beschreibungen zu den einzelnen Teilen des Pascal-Systems sind ein umfangreiches Nachschlagewerk, das Erfahrungen mit Pascal und dem UCSD-System voraussetzt.

Mit dem TI 99/4A macht es Spaß, Basic zu programmieren. Nach dem Motto »Schuster bleib bei Deinen Leisten« sollte man dem kleinen TI nicht unbedingt Pascal aufzwängen, zumal die Qualität des Systems den hohen Preis nicht rechtfertigt. (Gu)

ABS	180			POSITION (U)
ACCEPT	181	IF-THEN-ELSE	224	PRINT
ASC	183	IMAGE	226	PRINT USING
ATN	184	INIT (U)	228	
		INPUT	229	RANDOMIZE
BREAK	185	INT	231	READ
BYE (D)	187			REC
		JOYST (U)	232	REM
CALL	188			RES (D)
CHAR (U)	189	KEY (U)	233	RESEQUENCE (D)
CHARPAT (U)	192			RESTORE
CHARSET (U)	193	LEN	235	RETURN
CHR$	194	LET	236	RND
CLEAR (U)	195	LINK (U)	237	RPT$
CLOSE	196	LINPUT	238	RUN
COINC (U)	197	LIST (D)	239	
COLOR (U)	199	LOAD (U)	240	SAVE (D)
CON (D)	201	LOCATE (U)	241	SAY (U)
CONTINUE (D)	201	LOG	242	SCREEN (U)
COS	202			SEG$
		MAGNIFY (U)	243	SGN
DATA	203	MAX	245	SIN
DEF	204	MERGE (D)	246	SIZE (D)
DELETE	205	MIN	247	SOUND (U)
DELSPRITE (U)	206	MOTION (U)	248	SPGET (U)
DIM	207			SPRITE (U)
DISPLAY	208	NEW	249	SQR
DISPLAY USING	210	NEXT	250	STOP
DISTANCE (U)	211	NUM (D)	251	STR$
		NUMBER (D)	251	SUB
EDIT (D)	212			SUBEND
END	213	OLD (D)	252	SUBEXIT
EOF	214	ON BREAK	253	
ERR (U)	215	ON ERROR	254	TAB
EXP	216	ON-GOSUB	255	TAN
		ON-GOTO	256	TRACE
FILES (U)	217	ON WARNING	257	
FOR-TO-STEP	218	OPEN	258	UNBREAK
		OPTION BASE	261	UNTRACE
GCHAR (U)	220			
GOSUB	221	PATTERN (U)	262	VAL
GOTO	222	PEEK (U)	263	VCHAR (U)
		PI	264	VERSION (U)
HCHAR (U)	223	POS	265	

Bild 1. Befehlsliste des TI Extended Basic. »D« kennzeichnet Befehle des Betriebssystem und »U« Unterprogramme

Lehrbuch und Nachschlagewerk für TI Basic/ Extended Basic

Sämtliche Anweisungen, Unterprogramme und Funktionen des TI Basic und TI Extended Basic erklärt das Buch »TI Basic/Extended Basic für Anfänger und Fortgeschrittene« von Texas Instruments anschaulich und ausführlich.

Das Buch ist in zwei Teile gegliedert. Der erste führt den Leser schrittweise an das Programmieren in Basic heran. Es wird gezeigt, wie man den Computer benutzt oder einfache Programme mit dem Editor schreibt. Der Leser wird angeleitet grafische Muster in 16 Farben auf dem Bildschirm zu entwerfen sowie Töne, Melodien und Akkorde auszugeben. Im zweiten Teil werden sämtliche Befehle alphabetisch geordnet dargestellt. Damit ermöglicht dieses Buch nicht nur ei-

Fortsetzung auf Seite 155

Einfach schreiben mit dem TI-Writer

Angekündigt war er schon seit Herbst 1982, der TI-Writer. Um so gespannter darf man nun sein, was dieses Textverarbeitungsprogramm für den TI 99/4A alles kann.

Für den Einsatz des TI-Writers sind das Grundgerät (Konsole) TI 99/4A, die RAM-Erweiterung (Memory Expansion), der Floppy-Disk-Disketten-Controller mit mindestens einem Laufwerk, die Schnittstelle RS232 und ein Drucker nötig. Ob die TI-Peripherie in der früheren separaten Form oder als Steckkarten für die neue Erweiterungsbox vorliegt, ist dabei unerheblich. Lediglich die Konsole darf nicht vom alten Typ TI 99/4 sein.

Diese Ansprüche an die Gerätekonfiguration sollte man nicht vergessen, wenn Zusatzgeräte noch fehlen. Die Gesamtkosten der Anlage liegen dann mit dem TI-Writer grob geschätzt immerhin bei mindestens 4000 Mark (je nach Drucker) und fallen damit in eine Preisklasse, bei der die »kommerziellen« Mikrocomputer beginnen.

Ein leichtes Schwergewicht

Der TI-Writer besteht aus einem dicken Paket, welches ein ROM-Modul (»TI-Writer/Word Processor«), eine Diskette mit Software und ein Handbuch mit über 170 Seiten enthält. Letzteres ist äußerlich ebenso wie inhaltlich sauber gestaltet. Es enthält einen Übungs- und Einführungsteil mit einer Beschränkung auf das Wesentliche.

Allerdings unverzichtbare Voraussetzung für den Einstieg in die Materie sind derzeit noch gute Englischkenntnisse, da das Handbuch nur in dieser Sprache vorliegt. Aber vielleicht trägt man den gestiegenen Absatzzahlen in der Bundesrepublik durch die baldige Herausgabe einer deutschen Übersetzung des Handbuchs Rechnung. Dies könnte für den Markterfolg größere Bedeutung haben, als viele Seiten Anzeigen. Immerhin ist der TI-Writer zur Zeit wohl das erste echte Textverarbeitungssystem im Bereich der Heimcomputer — und das zu einem Preis von momentan zirka 550 Mark.

Leugnen zwecklos

Auch mit dem TI-Writer im Port kann der TI 99/4A seine Eigenschaft als Heimcomputer nicht verleugnen. Statt in eintönigem Weiß strahlt der Bildschirm in dunklem Blau. Soweit nicht schlecht, denn diese Kombination ist ein annehmbarer Kompromiß sowohl für Monitor als auch für den Bildschirm eines Fernsehgerätes. Außerdem stehen vier weitere Farbkombinationen zur Wahl.

Eine andere schon vom Zeichensatz für das TI-Basic her gewohnte Eigenheit des TI 99/4A fällt ins Auge, weil sie im Zusammenhang mit einem Textsystem eine nicht ganz verständliche Einschränkung der Benutzerfreundlichkeit darstellt. Auch bei der derzeitig im Handel befindlichen Version des TI-Writers sind die Kleinbuchstaben nichts weiter als gestauchte Großbuchstaben und daher allzuleicht verwechselbar. Daß es auch anders geht, zeigt der deutsche Zeichensatz: Das kleine ä erscheint in der richtigen Kleinform, im Gegensatz zum a (Bild 1, Zeile 19). Nach Auskunft des Herstellers soll allerdings noch dieses Jahr eine deutsche Version mit echter Kleinschreibung angeboten werden.

In diesem Zusammenhang darf nicht unerwähnt bleiben, daß der TI-Writer sechs nationale Zeichensätze bereitstellt, je nachdem, welche Sprache man im Eingangsmenü anwählt. Dies hat nicht einmal jedes professionelle System zu bieten.

Die Länge einer Bildschirmzeile ist nicht die gleiche wie im Basic-Bereich. Eine Bildschirmseite faßt im Textmodus des TI-Writers 24 x 40 Zeichen. Das ist nicht gewaltig und für Textverarbeitung wohl die allerunterste Grenze, aber ein TV-Schirm schafft dieses schon kaum mehr. Deshalb wird wohl jeder zukünftige Besitzer eines TI-Writers früher oder später seiner Augen zuliebe einen Monitor hinzukaufen. ►

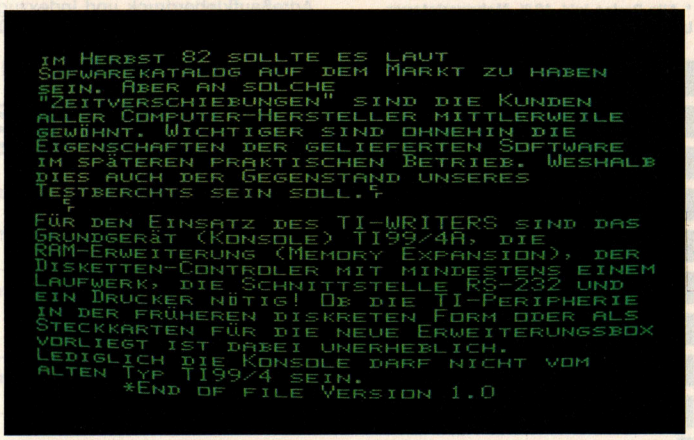


Bild 1. Die Zeilenbreite des Bildschirms beträgt 40 Zeichen. Ein gewöhnlicher Farbfernseher wird dabei zum Marterinstrument für die Augen.

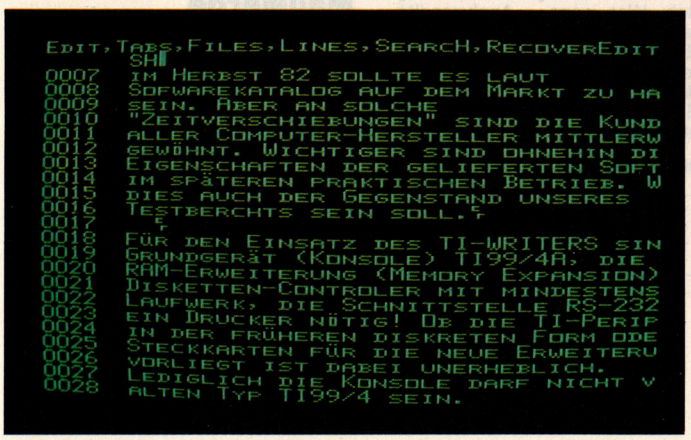


Bild 2. Selbst am einfachsten Monitor ist die Schrift klar und leicht zu lesen. So schafft man auch längere Texte ohne Augentropfen.

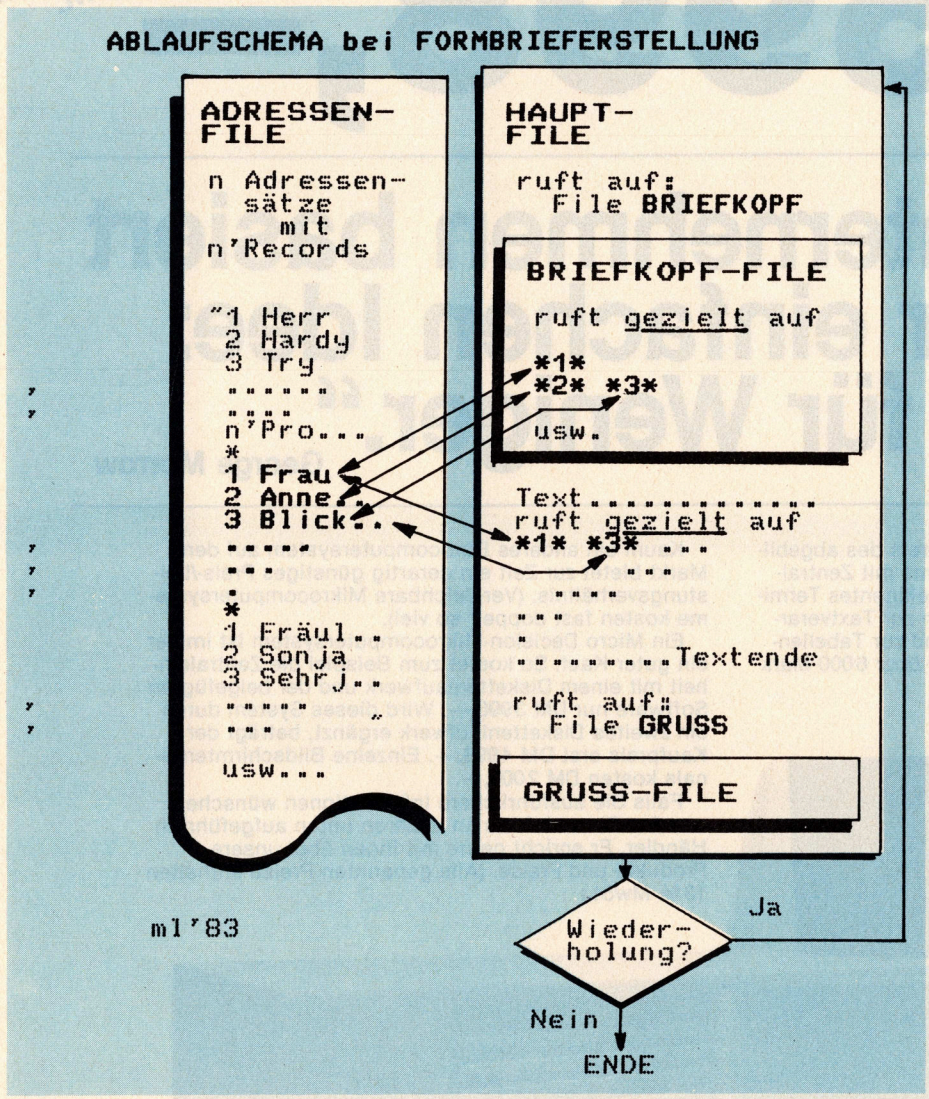


Bild 3. Ablaufschema bei der Erstellung von Formbriefen.

Es muß ja nicht gleich einer mit Farbe sein, ein grüner tut es auch (ab zirka 300 Mark; Bilder 1 und 2).

Die eigentliche Textzeile ist 80 Zeichen breit. Der Schirmausschnitt kann wie ein Fenster in drei sich überlappende Zonen von je 40 Zeichen Breite verschoben werden. Erreicht der Cursor den rechten oder linken Schirmrand und ist darüber hinaus noch Text vorhanden, springt dieses Fenster von selbst weiter. Dennoch ist es erfahrungsgemäß beim flüssigen Schreiben unangenehm, wenn man nicht einige Zeilen weit den Text geschlossen vor sich sieht. Praktischer ist es deshalb, mit dem Tabulator die Zeilenbreite auf die des Bildschirms einzuengen und die Länge der Druckzeile erst mit Hilfe des Formatierers (wird später noch beschrieben) einzustellen.

Ähnlich wie die Bildschirmausgabe ist die Kapazität des Arbeitsspeichers etwas eingeschränkt. Sie beträgt ungefähr 22600 Zeichen. Das entspricht sechs Briefseiten mit 60 Zeichen pro Zeile oder 3⅓ vollen Textseiten aus Computer persönlich. Für den häuslichen und den halbprofessionellen Gebrauch reicht das aus, zumal beim Ausdruck durch ein Hauptfile beliebig viele Textfiles von bis zu drei Diskettenstationen aneinandergereiht werden können. Auf diese Weise lassen sich ohne Diskettenwechsel an die 240000 Zeichen auf das Papier bringen (das wären 36 Druckseiten dieser Zeitschrift).

Der Editor: Tor zum Text

Die Software des TI-Writers besteht aus zwei Teilprogrammen, die nur getrennt aufgerufen werden können. Das ist zum einen der Editor und zum anderen der Formatierer. Jedes dieser Programme hat eine grundverschiedene Aufgabe zu erledigen, aber manches Endergebnis kann mit beiden erreicht werden.

Der Editor gestattet dem Benutzer die Eingabe neuer Texte und den Zugang zu den schon gespeicherten. Er stellt also sowas wie ein Tor zum Textspeicher dar, den er gleichzeitig verwaltet. Sein »Zeigestock« ist der Cursor, sein Kommunikationsmittel mit dem Anwender die Kommandozeile.

Editorprogramme können so ausgelegt sein, daß sie ein sehr direktes Arbeiten mit dem Cursor gestatten, in der Art etwa, wie man ein Video-Raumschiff über den Bildschirm lenken kann. Dann nennt man sie »bildschirmorientiert«. Oder sie sind mit umfangreichen Verwaltungsbefehlen versehen (befehlsorientiert).

Praktisch alle Arbeiten können sowohl mit dem einen wie mit dem anderen Editortyp erledigt werden. Welche Lösung die bessere ist, hängt von der Art der Aufgabe ebenso wie vom persönlichen Geschmack und Arbeitsstil des Anwenders ab. Die meisten Menschen sind allerdings »Augen«-Menschen und bevorzugen den bildschirmorientierten Editortyp. Seine Bedienung ist zudem viel schneller zu erlernen. Daß TI diese Art Editor wählte, darf also nicht verwundern, zudem auch sein Befehlsvorrat keinen Vergleich zu scheuen braucht.

Für die Texteingabe stehen zwei Betriebsarten zur Verfügung, der Word-Wrap-Modus und der Fixed-Modus. Am Bildschirm werden sie durch unterschiedliche Cursorformen angezeigt.

Der Word-Wrap-Modus bietet sich für die flüssige Eingabe durchgehender Texte an. Dabei werden Worte, die am Ende der jeweiligen Zeile nicht mehr ganz unterzubringen sind, automatisch in die nächste Zeile übertragen. Erst am Ende eines Absatzes muß die Enter-Taste gedrückt werden. Bei nachträglichen Einfügungen im Text wird die Zeile an der Stelle, wo der Cursor gerade steht, in eine obere, mit dem links vom Cursor stehenden Textteil und eine untere, mit dem Rest der ursprünglichen Zeile gesplittet. Jetzt kann bis zum Ende der Einfügung weitergeschrieben werden.

Zum Wiederanfügen des Resttextes drückt man einfach die Taste Reformat, und der folgende Text wird bis zum nächsten Wagenrücklaufzeichen oder dem Fileende neu formatiert, das heißt mit dem vorhandenen Text verbunden und

gleichzeitig umgebrochen (deshalb »Word-Wrap«).

Der Fixed-Mode hingegen kennt weder Umbruch noch Wagenrücklauf. Bei Einfügungen wird lediglich der rechte Teil der Zeile verschoben, um dem neu Eingetippten Platz zu machen. Was über den rechten Zeilenrand hinausgeht, wird gelöscht. Außerdem muß jede Zeile mit Enter abgeschlossen werden, ehe in der neuen Zeile weitergeschrieben werden kann.

Der Anwendungsbereich dieses Modus liegt im allgemeinen eher im Erstellen von Tabellen, Texten also, bei denen eine stärkere Kontrolle der formalen Verteilung nötig ist, obgleich der Übergang der beiden Anwendungsbereiche nicht starr gesehen werden darf. Die Umschaltung ist jederzeit möglich. Die Grafik zur Formbrieferstellung (Bild 3) wurde zum Beispiel im Fixed-Mode erstellt.

Allerhand Kommandos

Eine Reihe von Kommandos bedarf keiner besonderen Erörterung, da sie von anderen Textsystemen her gut bekannt sind. Sie gehören zum Standard aller Textsysteme. Dazu zählen diverse Tabulator-, String- und Copy-Funktionen; die Zahl der implementierten Funktionen ist sehr groß (zirka 30 Editorfunktionen, die Variationen nicht mitgezählt). Allein bei der Cursorsteuerung gibt es außer den altbekannten noch ein Dutzend weitere.

Eine ganze Kommandogruppe dient dem Bereich des File-Handlings. Die einzelnen Kommandos können praktisch beliebig gemischt werden. Eine bemerkenswert komfortable Funktion darf übrigens in diesem Zusammenhang nicht unerwähnt bleiben: Show Directory. Nach diesem Kommando listet der TI-Writer das Inhaltsverzeichnis der bezeichneten Diskette nach dem Vorbild des Disk Manager-Moduls auf. Auch mitten in der Texteingabe darf dergestalt die Neugier befriedigt werden, ohne daß Textdaten oder der jeweilige Eingabestatus verändert werden.

Besonders sympathisch sind einige Hilfsfunktionen gegen versehentliches Löschen oder Überschreiben von Textteilen. Es scheint, als hätten die TI-Leute einen Beschützerinstinkt für notorische Tastenmuffel. Bedingt durch die Enge und Mehrfachbelegung des Keyboards wird solcher Irrtum allerdings auch besonders gefördert (Insert-Taste und Delete-Line-Taste).

Mit Hilfe der sogenannten Oops!-Taste kann nämlich die gelöschte oder überschriebene Zeile wieder sichtbar gemacht werden, vorausgesetzt zwei Bedingungen sind erfüllt: Es darf erstens zwischen der Änderung und dem Oops!-Aufruf keine andere Funktion (auch nicht Enter) in Anspruch genommen und zweitens noch keine neue Zeile erreicht worden sein — trotzdem eine feine Sache.

Die zweite in diesem Zusammenhang erwähnenswerte Funktion ist »Recover Edit«. Angenommen Sie haben vor dem Ausstieg aus dem Editorteil das Abspeichern des Textes auf Diskette vergessen ... Zugegeben, dazu gehört eine große Portion Schusseligkeit. Immerhin waren drei verschiedene Kommandos nötig, um so weit zu kommen. Aber mir ist es trotzdem passiert.

Da hilft anderswo nichts mehr, außer neu tippen — nicht so beim TI-Writer. Man wählt wieder den Editor an, drückt RE für »Recover Edit«, und siehe da, der Text ist an seinem warmgehaltenen Platz (»warm« muß er schon noch sein; nach dem Ausschalten des Computers oder dem Entnehmen des TI-Writer-Moduls läßt sich nämlich gar nichts mehr »wiederentdecken«). Nur die ersten Zeilen fehlen. Aber das nimmt man in so einem Fall mit seligem Lächeln in Kauf.

Eine weitere Besonderheit ist der Special Character-Modus. Durch Umschalten des Cursors mit CTRL U können Steuerzeichen in den Text eingefügt werden, die normalerweise durch die Tastatur nicht ansprechbar sind. Am Schirm erscheinen sie als Kleinbuchstaben oder Zahlen mit Index.

Die Eingabe etwa der Hälfte der Kommandos erfolgt über eine Tastensequenz aus einer bestimmten

```
Herr
Hardy Try
Käuferweg 83

D-0007 Probedorf

                                        München, 01.06.83


Sehr geehrter Herr Try,

   anbei erhalten Sie einen Formbrief, der mit dem neuen
TI-Writer geschrieben worden ist. Da für Sie, Herr Try
dieses Textsystem privat und beruflich von einigem Interesse
sein dürfte, möchte ich Ihnen den Testbericht in COMPUTER
PERSöNLICH empfehlen.

Mit freundlichen Grüßen

-------------------------------------------------------------

Frau
Anneliese Blickauchdurch
Überall-lstraße 19

9999 Verbraucherstadt 3

                                        München, 1.Juni 1983


Sehr geehrte Frau Blickauchdurch,

   anbei erhalten Sie einen Formbrief, der mit dem neuen
TI-Writer geschrieben worden ist. Da für Sie, Frau
Blickauchdurch dieses Textsystem privat und beruflich von
einigem Interesse sein dürfte, möchte ich Ihnen den
Testbericht in COMPUTER PERSöNLICH empfehlen.

-------------------------------------------------------------

Fräulein
Sonja Sehrjung
Teenagerallee 18

Abiweiler/b.Schulhausen

                                        München, den 1.6.1983


Sehr geehrtes Fräulein Sehrjung,

   anbei erhalten Sie einen Formbrief, der mit dem neuen
TI-Writer geschrieben worden ist. Da für Sie, Fräulein
Sehrjung dieses Textsystem privat und beruflich von einigem
Interesse sein dürfte, möchte ich Ihnen den Testbericht in
COMPUTER PERSöNLICH empfehlen.
```

Bild 4. Mit der Fill-Anweisung erhält man beim Formatieren des Grundtextes (vergleiche Bild 9) einen rechtsbündigen Ausdruck

Taste zusammen mit der CTRL- oder der FTCN-Taste. Die anderen Befehle sind über den Command-Mode zu erreichen, das bedeutet, daß zuerst mit FTCN 9 die Kommando-Eingabezeile am oberen Schirmrand aufgerufen werden muß. Dann wird eine Buchstabenkombination eingegeben und Enter gedrückt. Jetzt erst erfolgt die Ausführung des Kommandos. Wie man sieht, ist das etwas umständlich und schmälert den Bedienungskomfort erheblich. Hier sind aber die Grenzen ganz deutlich durch die Hardware vorgegeben.

Tastensalat

Gut gelöst ist angesichts dieser Einschränkung die Auflistung der Kommandos in der obersten Schirmzeile (Bild 2). Jeweils mehrere Befehle können durch einen Oberbegriff aufgerufen werden (zum Beispiel mit Search die Befehle Replace String und Find String). Wer allerdings das richtige Kommando schon auswendig weiß, dem bleibt die sofortige Eingabe unbenommen. Das Fehlen einer Hilfe-Taste für etwas ausführlichere Anweisungen ist dennoch schade. Wer garantiert, daß die mitgelieferte Quick Reference Card in einem Jahr noch auffindbar ist?

Ein Formatierer von Format

Wenn der Text fertig ist, geht es an den Ausdruck. Für diesen Part steht uns der Formatierer zur Verfügung. Sein Aufgabengebiet umfaßt alles, was mit der Form des zu Papier gebrachten Textes zu tun hat. Dabei bleibt das eigentliche Textfile unverändert. Seine Anweisungen erhält der TI 99/4A auf zweifache Weise: Bestimmte Daten, wie solche über die Ausgabeeinheit, Filename, Anzahl der Kopien etc. werden mittels Bildschirm vom Anwender erfragt. Die meisten Kommandos erhält er jedoch aus speziellen Zeilen im File, die bereits während der Texterstellung mit dem Editor eingefügt wurden und alle mit einem Punkt beginnen. In solchen Zeilen können die Parameter für Randbreiten, Zeilenabstand, Einrückungen und Zentrierungen, Seitenlängen und -wechsel definiert werden. Eine Kommandozeile der Form ».CO text...« dient dazu, für spätere Arbeiten mit dem File Kommentare zu

```
0001  .CO File: PRINTCOMM
0002  .CO TEST für COMPUTER PERSöNLICH
0003  .CO (Kommandos f.HONEYWELL S 11)
0004  .CO ============================
0005  .CO 6ZEILEN/ZOLL:
0006  .TL 3:27,55,48,27,54,48,48,55,48
0007  .CO 8ZEILEN/ZOLL:
0008  .TL 9:27,55,49,27,54,48,48,57,51
0009  .CO SPERRSCHRIFT EIN:
0010  .TL 2:27,56,53
0011  .CO SPERRSCHRIFT AUS:
0012  .TL 14:27,56,54
0013  .CO KOMPRIMIERTE SCHRIFT:
0014  .TL 17:27,55,53
0015  .CO NORMALE^SCHRIFT:
0016  .TL 23:27,55,51
0017  .CO UNTERSTREICHUNG EIN:
0018  .TL 15:18
0019  .CO UNTERSTREICHUNG AUS:
0020  .TL 16:20
0021  .CO ============================
```

Bild 5. Beispiel für ein druckerspezifisches File, das die Kommandosequenzen für den Drucker aus selbstdefinierten Zeichen im Textfile generiert.

```
0001  .CO File: BRIEFKOPF
0002  .CO TEST für COMPUTER PERSöNLICH
0003  .CO ============================
0004  .DP 9:DATUM:
0005  Achim Lenagl^^^^^^^^^^^^^^^^^^^^
0006  Tel.:^089/7271797
0007  .IN +44
0008  Farmerstraße 77
0009  D-8000 München 40
0010  .IN +0
0011  #^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^#
0012
0013
0014  *1*
0015  *2* *3*
0016  *4*
0017
0018  *5* *6*
0019
0020  .IN +38
0021  München, *9*
0022  .IN +0
0023
0024
0025  Sehr geehrte*11* *1* *3*,
0026
0027  .AD;IN +3
0028  .CO ============================
```

Bild 6. Briefkopf-File. So könnte das Formfile für einen gemäßigt grafisch gestalteten Briefkopf aussehen. Alle »*n*«-Stellen werden beim Druck durch Daten aus einer Mailing List aufgefüllt. *9* muß vom Bediener per Bildschirm mit dem aktuellen Datum ausgefüllt werden. Der Fragetext ist in Zeile 0004 definiert.

```
0001  .CO File: GRUSS
0002  .CO TEST für COMPUTER PERSöNLICH
0003  .CO ============================
0004
0005  .IN +0
0006  Mit freundlichen Grüßen
0007  .CO ============================
```

Bild 7. File für die Grußformel am Briefende.

ermöglichen. Sie hat keinen Einfluß auf den Ausdruck.

Die meisten der zwei Dutzend Funktionen sind mehr oder minder bei allen Textsystemen ähnlich und bedürfen keiner besonderen Erwähnung, einige aber sind zumindest bei der Geräte- und Preisklasse des TI-Writer recht ungewöhnlich. Darunter fällt zum Beispiel die Möglichkeit, automatische Kopf- und Fußzeilen mit oder ohne Text, mit oder ohne fortlaufender Seitenzahl oder beides gemischt generieren zu können. Die Seitenzählung läßt sich außerdem noch nach Bedarf im Text justieren.

Steht eine Fill-Anweisung im Text, so werden die Zeilen mit so vielen Wörtern wie möglich gefüllt; mit dem Adjust-Befehl fügt der Formatierer in den Text einer Zeile solange Leerzeichen ein, bis der rechte Rand bündig ist (Bild 4). Aber Vorsicht, der Formatierer geht von gleichbleibenden Buchstabenbreiten aus. Mit Proportionalschrift ergibt das unschönen Flatterrand.

Jede dieser Anweisungen kann übrigens für bestimmte Zeilen, Textabschnitte oder auch den ganzen Resttext außer Kraft gesetzt werden. Dies erhöht die ohnehin recht ansehnliche Flexibilität des TI-Wirter erheblich.

Kommandos nach Herzenslust

Eine feine Sache ist der Befehl ».TL n:m1, m2...«. Mit ihm kann jedes beliebige Zeichen mit dem ASCII-Wert n in eine neue Zeichenfolge umgewandelt werden, deren Zeichen die ASCII-Werte m1, m2 und so weiter besitzen. Erscheint dann im Text das umdefinierte Zeichen, sendet der Formatierer die Ersatzsequenz an den Drucker. Damit lassen sich (zusammen mit dem Special Character Mode des Editors) die Kommandoforderungen selbst exotischer Drucker befriedigen. Sinnvollerweise faßt man diese Zeilen in einem eigenen File zusammen, das wie in Bild 5 aussehen kann. Zeilen der Form ».Co ...« sind Kommentare. Die vorangestellten Zeilennummern wurden beim Ausdruck hinzugefügt. Jede Zeile schließt mit einem im Ausdruck nicht sichtbaren Wagenrücklauf. Am Anfang jedes Textfiles steht dann nur noch eine Referenz auf dieses Sonderfile. Das spart Diskettenspeicherplatz.

Keine eigene Kommandozeile braucht eine Gruppe von Befehlen, ►

die der Unterstreichung oder Hervorhebung durch mehrmaligen Anschlag dienen. Da sie einzelne Wörter oder Sätze betreffen, werden sie unmittelbar im Text den bezeichneten Satzteilen vorangestellt.

Die Anweisungen für das Einfügen von Daten aus sogenannten Mailing Lists (*n*) stehen zwar auch innerhalb des Textes, sind aber im Zusammenhang mit dem Kapitel über den Formbrief eine eigene Erörterung wert.

Hilfe — beispielsweise bei der Erstellung von Rundschreiben

Die vielleicht komplexeste Anwendung eines Textsystems ist die Erstellung von Formbriefen und deren Ausdruck mit diversen Adresseneinfügungen (vergleiche Bild 4). Eine Aufgabe, die im häuslichen Bereich sicher selten anstehen dürfte. Immerhin zielt man mit seinem Textsystem auch auf den Markt der Freiberufler und kleinen Gewerbetreibenden, bei denen solche Schreibarbeiten sehr wohl vorkommen können.

Zur Veranschaulichung der diversen Möglichkeiten für Formbriefe und deren Ausdruck sei auf Bild 6 bis 9 verwiesen. Gerade diese Vielfalt hebt den TI-Writer aus der Reihe der anderen Heimtextsysteme deutlich hervor.

Ein paar Hinweise zum Datenformat: Adreßfiles (Bild 10) müssen wie alle Files für den TI-Writer das Format »Display, Variable 80« besitzen. Am Ende jeder Zeile befindet sich das Wagenrücklaufzeichen mit dem ASCII-Wert 141, welches der Drucker im abgebildeten Listing natürlich nicht anzeigt. Zwischen den Datensätzen, die aus maximal 99 einzelnen Zeilen bestehen dürfen, steht jeweils ein Sternchen als Trennzeichen. Die bis zu zweistelligen Zahlen am Zeilenanfang sind der Indikator dafür, an welcher mit »*n*« gekennzeichneten Stelle im Text die Zeile später eingefügt werden soll. Eine bestimmte Reihenfolge ist nicht vorgeschrieben. Bild 10 zeigt diese Eigenheiten.

An den Geldbeutel rührt der Umstand, daß Texte unmittelbar nur auf den teuren Disketten abgespeichert werden können, nicht aber auf Kassetten. Im Bürobereich wäre das keine Diskussion wert. Kassetten sind sicher ein inakzeptables Medium für Textverarbeitung. Im Heimbereich läßt es sich aber

```
0001  .CO File: TESTTEXT
0002  .CO TEST für COMPUTER PERSöNLICH
0003  .CO ============================
0004  .IF DSK1.PRINTCOMM
0005  .ML DSK1.ADRESSEN
0006
0007  .FI;LM 10;RM 70
0008  .IF DSK1.BRIEFKOPF
0009     ************************
0010  .CO * HIER TEXT EINFüGEN ! *
0011     ************************
0012  .IF DSK1.GRUSS
0013  .CO ============================
```

Bild 8. Dieses File nimmt den eigentlichen Haupttext auf. Es enthält bereits alle die weiteren Formfiles verbindenden Kommandos.

```
0001  .CO File: TESTTEXT (ausgefüllt)
0002  .CO TEST für COMPUTER PERSöNLICH
0003  .CO ============================
0004  .IF DSK1.PRINTCOMM
0005  .ML DSK1.ADRESSEN
0006
0007  .FI;LM 10;RM 70
0008  .IF DSK1.BRIEFKOPF
0009  anbei erhalten Sie einen
0010  Formbrief, der mit dem neuen
0011  @TI-Writer geschrieben worden
0012  ist. Da für Sie, *1* *3* dieses
0013  Textsystem privat und beruflich
0014  von einigem Interesse sein
0015  dürfte, möchte ich Ihnen den
0016  &Testbericht in @COMPUTER
0017  @PERSöNLICH empfehlen.
0018  .IF DSK1.GRUSS
0019  .CO ============================
```

Bild 9. So sieht dieses Formfile ausgefüllt aus. Es erzeugt mit Hilfe des Formatiereres den Brief aus Bild 4.

```
0001  .CO File: ADRESSEN
0002  .CO TEST für COMPUTER PERSöNLICH
0003  .CO ============================
0004  .CO 3 DATENSäTZE
0005  1 Herr
0006  11 r
0007  2 Hardy
0008  3 Try
0009  4 Käuferweg 83
0010  5 D-0007
0011  6 Probedorf
0012  *
0013  1 Frau
0014  11
0015  2 Anneliese
0016  3 Blickauchdurch
0017  4 überallstraße 19
0018  5 9999
0019  6 Verbraucherstadt 3
0020  *
0021  1 Fräulein
0022  11 s
0023  2 Sonja
0024  3 Sehrjung
0025  4 Teenagerallee 18
0026  5
0027  6 Abiweiler/b.Schulhausen
0028  *
0029  .CO USW. ======================
```

Bild 10. Eine Mailing List. Die Sternchen trennen jeweils zusammengehörende Datensätze von den nächsten. Jeder Satz darf maximal 99 Zeilen beinhalten.

durchaus denken, daß der eine oder andere ein Formfile für die jährliche Steuererklärung lieber auf billiges Band verbannen möchte, als dafür teure Disketten opfern zu müssen. Auch die unbedingt zu empfehlenden Sicherheitskopien wären auf Band gut aufgehoben. Im Haushalt haben solche Preisgrößen nun mal ein ganz anderes Gewicht als im Profibereich. Eine Filekonvertierung und -übertragung von Diskette auf Kassette mittels eines Basic-Hilfsprogramms ist natürlich möglich, aber umständlich.

Black Box, High Light

Ich muß zugeben, daß während des Tests in mir die Bewunderung für diese kleine schwarze Zigarettenschachtel ständig stieg. Mit diesem Modul hielt ich ein hochwertiges Werkzeug in Händen, kein Spielzeug! Wer sich den TI-Writer zulegt, braucht vor der Umwelt — aber auch der häuslichen — keine Rechtfertigung darüber abzugeben, wozu sowas denn gut sei (die gefürchtetste Frage für alle computerbegeisterten Ehemänner). Nach spätestens zehn Minuten Vorführen im Familien- und Freundeskreis ist da nur noch Staunen zu sehen.

Auch wer kommerzielle Textsysteme vom beruflichen Bereich her kennt, staunt, was an Möglichkeiten in diesem kleinen und relativ preiswerten TI-Writer steckt. Das ist tatsächlich mehr, als in manchem »Billigpaket« für das Büro zum fünffachen Preis geboten wird.

Wohlgemerkt: Dieses Lob bezieht sich auf den TI-Writer, nicht auf den TI 99/4A. Der entspricht nämlich keineswegs den Anforderungen an professionellen Gebrauch, angefangen von der viel zu kleinen Tastatur mit teils recht absonderlich angeordneten Tastenfunktionen, bis hin zu der Beschränkung auf 40 Zeichen pro Zeile am Bildschirm. Überhaupt hat man den Eindruck, daß hier die Grenzen der Nützlichkeit einzig vom Gerät und nicht vom Programm gezogen werden. Der TI-Writer könnte ohne weiteres — auf einer größeren Maschine eingesetzt — auch berufliche Bedürfnisse befriedigen.

Das Gebotene ist sein Geld mehrfach wert. Eine Aussage, die am Markt nicht oft gemacht werden kann. Das Urteil über den tatsächlichen praktischen Nutzen aber muß natürlich jeder selbst für seine eigene Situation fällen. (M. Lang)

Funktionenplot am Bildschirm

Mit dem vorliegenden Programm können am Bildschirm Funktionen mit 160 x 160 Punkten dargestellt werden. Dies ist beim TI 99/4A insofern schwierig, weil kein direkter Zugriff auf den Bildschirmspeicher möglich ist.

Das Programm (siehe Listing) gliedert sich in zwei Abschnitte. Die Zeilen kleiner 100 dienen der Programmvorstellung und der Erläuterung, in welcher Form die darzustellende Funktion in die Zeile 100 zu schreiben ist.

Auf die Variablendimensionierung folgt der Eingabeblock bis Zeile 160, der TI 99/4A bestimmt daraufhin den notwendigen Ordinatenbereich. Für 20 Bildspalten wird ab Zeile 210 festgelegt, ob sich ein Teil der Kurve in den durch Zeilen und Spalten definierten Bildelementen befindet. Zur Rechenzeitverkürzung werden lediglich in Programmzeile 250 Grafikwerte gesucht. Die Abtastung des betreffenden Bildelementes erfolgt durch Schleifen (270 bis 310). Falls sich das Bildelement als leer herausstellt, wird zur nächsten Zeile weitergegangen.

Die Zeilen 330 bis 400 stellen aus der 8 x 8-Matrix den zur Grafikgestaltung nötigen Hex-Ausdruck her, mit dessen Hilfe in Zeile 410 ein Charakter umcodiert und zur Anzeige gebracht wird. Benützt werden dabei die Codes 91 bis 143; da höhere Codes im verwendeten Extended Basic nicht akzeptiert werden, erfolgt gegebenenfalls die Verwendung von Codes ab 33 bis 46. Für extrem steil verlaufende Kurvenzüge könnte zwar zwischen den auseinander wandernden Bildpunkten linear interpoliert werden, aber auf diese Möglichkeit wurde im vorliegenden Programm mit Rücksicht auf die bereits in dieser Form gegebene lange Rechenzeit verzichtet.

(Karl Hagenbüchner)

```
10 REM FUNKTIONENPLOTT AM BILDSCHIRM(hochaufloesend)
20 REM BY Karl Hagenbuchner,Haidfeldstr. 2, A-4050 TRAUN
30 CALL CLEAR :: CALL SCREEN(11)
40 DISPLAY AT(8,8):"FUNKTIONENPLOTT" :: DISPLAY AT(15,2):" Die abzubildende Funk
tion          ist in der Form"
50 DISPLAY AT(18,2):"     100 DEF Y=FUNKTION(X)           einzuschreiben."
60 DISPLAY AT(23,1):"Abbruch mit CLEAR, Neustart        mit RUN 100 !"
70 GOTO 70
100 DEF Y=SIN(X)+SIN(6*X)/5
110 OPTION BASE 1
120 DIM M$(8),V(8,8),YY(8)
130 CALL CLEAR :: CALL SCREEN(11):: FOR I=1 TO 14 :: CALL COLOR(I,2,11):: NEXT I
140 DISPLAY AT(2,1):"X-START:" :: ACCEPT AT(2,9)BEEP VALIDATE(NUMERIC):XA
150 DISPLAY AT(2,17):"X-ENDE:" :: ACCEPT AT(2,24)BEEP VALIDATE(NUMERIC):XE
160 DISPLAY AT(1,10):"BITTE WARTEN"
170 XB=XE-XA :: X=XA :: YMA=Y :: YMI=Y
180 FOR X=XA TO XE STEP XB/20 :: IF YMA<Y THEN YMA=Y ELSE IF YMI>Y THEN YMI=Y
190 NEXT X
200 YB=YMA-YMI
210 FOR SP=6 TO 25 :: I=1 :: GOSUB 470
220 YY(I)=Y :: YMAX=Y :: YMIN=Y :: FOR I=2 TO 8 :: GOSUB 470
230 IF YMAX<Y THEN YMAX=Y ELSE IF YMIN>Y THEN YMIN=Y
240 YY(I)=Y :: NEXT I
250 Z1=INT((24*YB-20*ABS(YMI)-20*YMAX)/YB)+1
260 FOR Z=Z1-2 TO Z1+4
270 FOR I=1 TO 8 :: FOR J=1 TO 8
280 YO=(24*YB-20*ABS(YMI)-(Z-1)*YB)/20-J*YB/160+YB/320
290 YU=(24*YB-20*ABS(YMI)-(Z-1)*YB)/20-J*YB/160-YB/320
300 IF YY(I)<=YO AND YY(I)>=YU THEN V(I,J)=1 ELSE V(I,J)=0
310 P=P+V(I,J):: NEXT J :: NEXT I
320 IF P=0 THEN 420 ELSE P=0
330 FOR I=1 TO 8 :: FOR J=1 TO 8 :: M$(I)=M$(I)&STR$(V(J,I)):: NEXT J :: NEXT I
340 FOR I=1 TO 8 :: FOR J=1 TO 4
350 I$=SEG$(M$(I),J,1):: IF I$="1" THEN K=K+2^(4-J)
360 NEXT J :: IF K>9 THEN H$=CHR$(55+K)ELSE H$=STR$(K)
370 K=0 :: N$=N$&H$
380 FOR J=5 TO 8 :: I$=SEG$(M$(I),J,1):: IF I$="1" THEN K=K+2^(8-J)
390 NEXT J :: IF K>9 THEN H$=CHR$(55+K)ELSE H$=STR$(K)
400 K=0 :: N$=N$&H$ :: M$(I)="" :: NEXT I
410 CALL CHAR(91+L,N$):: CALL HCHAR(Z-1,SP,91+L):: L=L+1 :: IF L=53 THEN L=-58 E
LSE IF L=-44 THEN 440
420 N$="" :: NEXT Z
430 NEXT SP
440 DISPLAY AT(1,10):RPT$(" ",12)
450 CALL SOUND(500,440,1)
460 GOTO 460
470 X=XA+(SP-6)*XB/20+I*XB/160-XB/320 :: RETURN
```

TI 99/4A

Listing des Programms »Funktionenplot am Bildschirm«

Der wichtigste Massenspeicher

Der derzeit bei Personal Computern wichtigste Massenspeicher ist das Floppy-Disk- oder Disketten-Laufwerk. Als Datenträger dient dabei eine magnetisch beschichtete, dünne, flexible Kunststoffscheibe, auf der sich in herkömmlicher Technik bei 5,25 Zoll Durchmesser bis etwa 1 MByte, bei 8 Zoll Durchmesser bis etwa 2 MByte speichern lassen. Die Laufwerke sind kompakt, bieten wahlfreien Zugriff auf die Daten sowie eine relativ hohe Speicherkapazität bei zumeist ausreichend niedriger mittlerer Zugriffszeit — und das alles zu einem relativ günstigen Preis.

Die Datenträger (Disketten) sind zwar teurer als Musikkassetten, Diskettenlaufwerke deutlich teurer als die Audiokassettenrecorder (beim Vergleich mit professionellen Digitalkassetten-Laufwerken sieht die Sache anders aus). Dafür sind sie in der Regel erheblich schneller, und sie erlauben den wahlfreien Zugriff (Random Access): Während ein Band immer erst bis zu der Stelle vor- oder zurücklaufen muß, an der die gesuchte Information gespeichert ist, kann bei der Floppy-Disk auf den jeweils gewünschten Datensatz direkt zugegriffen werden.

Ihre Überlegenheit am deutlichsten zeigt die Diskette da, wo schnell hintereinander Daten benötigt werden, die an physikalisch unterschiedlichen Stellen gespeichert sind, also wenn Sie beispielsweise einen Rabattsatz für Einkaufsgenossenschaften, die Adresse eines Vertreters, den Preis vom Modell 13 und außerdem noch den Liefertermin einer gerade angekündigten Neuheit wissen wollen (das kann während eines Telefongesprächs oder genauso gut beim Fakturieren sein). Wenn das auf einem Band gesucht werden soll, wird es kaum jemand abwarten wollen.

Es ergeben sich, verglichem mit der Aufzeichnung auf Magnetband, eine ganze Reihe von Vorteilen: Die relativ langen Zeiten für das Suchen eines auf Kassette gespeicherten Programmes oder Datensatzes entfallen fast ganz (was sich allerdings in gewissem Umfang auch dadurch umgehen läßt, daß man auf Kassetten mit kurzen Bändern pro Spur jeweils nur ein Programm speichert); die Zeiten zum Einlesen beispielsweise eines Programmes vom Speicher in den Rechner können von Minuten auf Sekunden sinken; Einfügungen sind ohne weiteres möglich; durch Löschungen freigewordener Platz läßt sich sofort und vollständig wieder anders nutzen.

Zur Erläuterung der Unterschiede hat Lindsay Doyle in der englischen Fachzeitschrift Printout folgenden Vergleich veröffentlicht: Zum Test wurde ein 6 KByte langes Programm so oft als möglich einmal mit Hilfe eines Commodore-Kassettenlaufwerks auf einer Seite einer C12-Kassette, zum anderen auf einer Floppy-Disk mit Hilfe des Commodore-Laufwerks 3040 gespeichert (maximale theoretische Speicherkapazität zirka 20 KByte beim Band und 176 KByte bei der Diskette). Als kürzeste und längste Zeit zum Laden des Programmes vom Speicher in den Rechner wurden bei der Floppy 3,9 und 7,7 Sekunden, beim Band 123 und 371 Sekunden ermittelt.

Die eigentliche Diskette ist eine dünne, mit magnetisierbarem Material beschichtete Scheibe aus Kunststoffolie. Diese Scheibe ist in eine quadratische Schutzhülle eingebettet, in der sie sich, angetrieben vom Floppy-Laufwerk, mehr als 300 mal pro Minute dreht. Der Datenträger ist in konzentrische Spuren (in der Regel 35 bei der Minifloppy mit 5,25 Zoll Durchmesser und 77 bei der 8-Zoll-Diskette) und

Speicherkapazität der 8"-und 5,25"-Disketten (in Bytes) BASF/82

Typ	Unformatiert (ca.-Angaben)	Beispiele für Formatierung	Bytes
5,25" 1X	125.000	16 x 128 Bytes/Sektor	
		(bei 35 Spuren)	71.680
		(bei 39 Spuren)	79.872
		9 x 256 Bytes/Sektor	
		(bei 35 Spuren)	80.640
		(bei 39 Spuren)	89.856
5,25" 1D	250.000	16 x 256 Bytes/Sektor	
		(bei 35 Spuren)	143.360
		(bei 39 Spuren)	159.744
5,25" 2D	500.000	16 x 256 Bytes/Sektor	
		(bei 35 Spuren)	286.720
		(bei 40 Spuren)	319.488
Sind generell 40 Spuren initialisiert (z. B. bei BASF), stehen 39 Datenspuren und eine Indexspur zur Verfügung.			
8" 1X	250.000	26 x 128 Bytes/Sektor	242.944
		15 x 256 Bytes/Sektor	284.160
		8 x 512 Bytes/Sektor	303.104
8" 1D	500.000	32 x 256 Bytes/Sektor	606.208
8" 2X	500.000	15 x 256 Bytes/Sektor	568.320
		26 x 128 Bytes/Sektor	485.888
		26 x 128 Bytes/Sektor	
		Flip-Over-Version	485.888
8" 2D	1.250.000	26 x 256 Bytes/Sektor	985.088
		8 x 1024 Bytes/Sektor	1.212.416
Sind generell 77 Spuren initialisiert (z. B. bei BASF), stehen 76 Datenspuren und eine Indexspur zur Verfügung.			

Speicherkapazität von Disketten an BASF-Beispielen. Es bedeuten: 1X = einseitig benutzbar mit einfacher Aufzeichnungsdichte in Laufwerken mit einem Schreib/-Lesekopf. 1D = Einseitig benutzbar mit doppelter Aufzeichnungsdichte in Laufwerken mit einem Kopf. 2X = Beidseitig benutzbar mit einfacher Aufzeichnungsdichte in Laufwerken mit einem Kopf. 2D = Beidseitig benutzbar mit doppelter Dickte in Laufwerken mit zwei Köpfen.

außerdem in tortenstückförmige Sektoren eingeteilt. Gespeichert werden jeweils Datensätze (Blöcke) von 128 Byte oder 256 Byte Länge. Jeder solcher Datensatz beziehungsweise Block ist durch Angabe von Spur- und Sektornummer adressierbar. Zum Speichern oder Auslesen einer Information wird der Schreib-/Lesekopf von einem Arm (an dem er befestigt ist) bis zur gewünschten Spur bewegt — der gesuchte Sektor wird durch Drehung der Diskette erreicht. Der Zeitbedarf zum Lesen (oder Schreiben) eines bestimmten Blockes liegt bei gängigen Laufwerken durchschnittlich im Bereich zwischen 100 ms und 350 ms (sogenannte mittlere Zugriffszeit).

Diskette ist nicht gleich Diskette

Diskette ist keineswegs gleich Diskette: Prinzipiell gibt es zwei Diskettentypen — hard- oder softsektorierte. Bei hardsektorierten Disketten werden die einzelnen Sektoren jeweils durch in den Datenträger eingestanzte Löcher gekennzeichnet. Diese Löcher werden optisch abgetastet und so die Sektornummer ausgewertet. Bei der Softsektorierung werden die einzelnen Sektoren durch einen speziellen »Daten-Vorspann« (Realisierung durch Software, Disketten-Betriebssystem) bestimmt, der Speicherplatz kostet. Diese Formatierung wird entweder vom Diskettenhersteller selbst (initialisierte Disketten) oder vom Anwender (nicht-initialisiert) vorgenommen. Die Ausnutzung der vorhandenen (unformatierten) Speicherkapazität ist der bei Hardsektorierung besser, das heißt die formatierte Speicherkapazität ist höher. Demgegenüber hat die Softsektorierung den Vorteil, daß mit ihr IBM-kompatible (8-Zoll) Disketten erzeugt werden können, was für eine Reihe von Anwendungen interessant sein kann.

Augenfälligster Unterschied zwischen den verschiedenen Diskettentypen sind die äußeren Abmessungen. Da gibt es zunächst — schon seit 1973 — die von IBM entwickelte 8-Zoll-Diskette. IBM hat mit dieser Diskette und dem entsprechenden Aufzeichnungsformat einen Industriestandard gesetzt. Für den Datenträgeraustausch (auch mit Großrechnern) ist sie immer noch am besten geeignet. Das hat beispielsweise dazu geführt, daß sehr viel CP/M-Software, die ja überwiegend auf Rechnern mit 5,25-Zoll-Floppy-Laufwerken eingesetzt wird, auf 8-Zoll-Disketten geliefert und dann vom Händler oder Systemhaus auf die Mini-Disketten des jeweiligen Kunden beziehungsweise Anwenders umgesetzt wird. Dem Vorteil sehr weitgehender Kompatibilität stehen zwei wesentliche Nachteile gegenüber: Die Laufwerke brauchen verhältnismäßig viel Platz, was vor allem bei Tischgeräten stört, und sie sind (das gilt auch für die Datenträger) erheblich teurer als Systeme, die mit kleineren Durchmessern arbeiten.

Die Mini-Diskette mit 5,25 Zoll Durchmesser ist vier Jahre jünger als die 8-Zoll-Diskette — inzwischen aber schon weiter verbreitet als ihre größere »Schwester«. Auf einer 5,25-Zoll-Diskette lassen sich heute — abhängig von verwendetem Rechner, Aufzeichnungsverfahren und Laufwerk — etwa zwischen 80 KByte und über 3 MByte unterbringen. Von der technisch möglichen Speicherkapazität her ist also die Mini-Diskette der »normalen« Diskette durchaus ebenbürtig; die Laufwerke sind aber sehr viel kompakter und billiger. Das hat zwar zu ihrer starken Verbreitung geführt — leider aber nicht zu einem einheitlichen Aufzeichnungsformat. So ist heute nicht einmal mehr gesagt, daß zumindest innerhalb des Programms eines Computerherstellers die Mini-Floppys ohne weiteres untereinander kompatibel sind. Da wird teils mit einfacher, teils mit doppelter Dichte aufgezeichnet, mal nur auf eine, mal auf beide Seiten der Diskette — und das alles natürlich mit unterschiedlichen Spur- und Sektorzahlen. Die Vielfalt der Kombinationsmöglichkeiten ist inzwischen so groß, daß die Diskettenhersteller zentimeterdicke Handbücher veröffentlichen, in denen man nachlesen kann, welcher Diskettentyp sich für welche Laufwerke und Rechner eignet.

Es kann nicht bestritten werden, daß sich die Hersteller um ein bestimmtes Maß an Kompatibilität bemühen. Das gilt beispielsweise für die äußeren Abmessungen der Datenträger und zum Teil auch der Laufwerke. Zum Teil basiert die Kompatibilität auch auf dem Bestreben, andere Produkte zu ersetzen. So stellte Philips kürzlich 5,25-Zoll-Laufwerke vor, die »kompatibel zu den bisher am Markt üblichen 8-Zoll-Laufwerken sind« (Speicherkapazität 1,6 MByte unformatiert bei doppelseitiger Speicherung).

Die Bemühung, alles immer kompakter zu machen, hat nicht nur dazu geführt, daß es jetzt beispielsweise von BASF 5,25-Zoll-Floppy-Laufwerke gibt, die nur noch eine Einbauhöhe von 33,5 mm (Doppellaufwerke: 57,5 mm) haben: Mit der Entwicklung von 3-, 3¼-, 3½- sowie 3,9-Zoll-Floppys samt den entsprechenden Laufwerken sind neue Inkompatibilitäten geschaffen worden — und zugleich neue Unsicherheiten. Bis heute weiß nämlich niemand exakt zu sagen, welches dieser Formate sich denn nun durchsetzen wird. Die 3,5-Zoll-Floppy scheint bislang einen gewissen Vorsprung zu haben; wie der Erfolg von IBM, die am Markt bisher schon vieles durchgesetzt hat, mit der 3,9-Zoll-Floppy sein wird, ist bislang völlig offen. Gegenüber den bislang bekannten Disketten, die auch eine flexible Hülle hatten, bieten die Mikro-Floppys (das sind die Formate unter 4 Zoll Durchmesser, auf denen sich bislang 250 KByte oder 500 KByte speichern lassen) den Vorteil, daß sie eine feste Hülle aus hartem Kunststoff haben. Sie

Laufwerke aus möglichst wenig Einzelteilen

sind damit gegen unachtsame Behandlung und mechanische Beschädigungen wesentlich besser geschützt und lassen sich andererseits aufgrund ihres Formates leicht auch in die Tasche stecken. Sie bieten sich nicht nur für tragbare Computer, sondern auch für kompakte Tischgeräte an.

Die angebotenen Floppy-Laufwerke unterscheiden sich intern im Aufbau im wesentlichen in drei Punkten: beim Spindelantrieb, bei der Positionierung des Armes mit dem Schreib/Lesekopf und im Einsatz von Kopflademagneten. Die Spindel, die die Floppy in Umdrehung versetzt, kann entweder direkt oder über Riemen angetrieben werden. Der Arm kann über einen Bandmechanismus mit einem Schrittmotor oder mit Hilfe einer Tauchspule positioniert werden. Der Schreib/Lesekopf kann entweder ständig auf der Diskette aufliegen oder aber bei Schreib/Leseoperationen mit Hilfe eines Lademagneten aufgesetzt (und anschließend wieder abgehoben) werden. Die Tendenz geht dahin, Laufwerke aus möglichst wenig Einzelteilen

aufzubauen, was nicht nur dem Preis, sondern auch der Zuverlässigkeit zugute kommt. Andererseits bemüht man sich darum, die Laufwerke so zu konstruieren, daß die Diskette möglichst wenig abgenutzt wird, wobei die Ansichten durchaus unterschiedlich sind, wie man das am besten erreicht: Es gibt Konstruktionen, bei denen die Diskette zunächst auf die Solldrehzahl hochläuft, dann der Kopf zum Schreiben beziehungswese Lesen aufgesetzt wird, danach wieder abhebt und die Diskette anschließend ausläuft. Bei einem anderen Konstruktionsprinzip dreht sich die Diskette ständig — der Kopf wird zum Lesen abgesenkt und anschließend wieder angehoben; bei einer dritten Konstruktion liegt der Kopf ständig auf der Diskette auf, die durch den Antrieb sehr schnell (etwa innerhalb einer Umdrehung) auf die Solldrehzahl gebracht und anschließend wieder gestoppt wird; die letztere Konstruktion arbeitet ohne Kopflademagneten.

Der normale Computeranwender, der nicht zum Eigenbau tendiert, ist in der Regel zunächst einmal auf das angewiesen, was ihm der Computerhersteller offeriert, sei es als externer Speicher, sei es als integraler Bestandteil des Computersystems. Zur Nutzung eines Floppy-Disk-Laufwerks ist nämlich außer dem reinen Laufwerk einmal ein Controller, zum anderen ein Disc Operating System (DOS) erforderlich. Das DOS hat unter anderem die Aufgabe, einen Index zu führen, zu berechnen, wieviel Speicherplatz belegt und wieviel noch frei ist, und zu registrieren, an welcher Stelle der nächste zu lesende Block steht, wenn beispielsweise ein gespeichertes Programm wegen seiner Länge mehrere Blöcke beansprucht. Das DOS kann Bestandteil eines Betriebssystems wie beispielsweise CP/M oder MS-DOS sein oder aber — wie es beispielsweise Commodore macht — in einen »intelligenten« Massenspeicher ausgelagert werden (was dann zu rechnerspezifischen Massenspeichern führt). Außerdem ist zum Betrieb ein auf den jeweiligen Computer zugeschnittener Controller erforderlich. Er wird in der Regel vom Rechnerhersteller, häufig auch von unabhängigen Firmen, insbesondere den Anbietern anschlußfertiger Speicher-Subsysteme und in den seltensten Fällen vom Laufwerkshersteller entwickelt und geliefert. Während Computerhersteller, Systemhäuser und andere Abnehmer mit technischem Know-how eine verhältnismäßig breite Auswahl an Floppy-Disk-Laufwerken haben, bleibt dem Computerbenutzer in der Regel nur die Wahl zwischen dem, was der Computerhersteller anbietet und dem, was unabhängige Hersteller als kompatibles, anschlußfertiges System offerieren. Wie groß oder klein dieses Angebot unabhängiger Hersteller ist, hängt nicht nur von der Verbreitung des jeweiligen Rechners, sondern auch vom konstruktiven Konzept ab.

Es ist bei Rechnern, die wie der Apple II oder der IBM-PC oder die S100 Bus-Systeme weitgehend nach dem Steckkartenprinzip aufgebaut sind, größer als beim Commodore. So gibt es beispielsweise für den Apple II nicht nur kompatible Floppy-Disk-Laufwerke mit einer Speicherkapazität von 143 KByte pro Diskette wie beim Original, sondern auch mit 164 und 286 KByte. Wer nicht zwei Einzellaufwerke anschließen will, bekommt von Comrex auch ein anschlußfertiges Doppellaufwerk (2 x 143 KByte) oder von Vista ein Laufwerk, das mit Kassetten arbeitet; in jeder Kassette sind fünf Disketten mit einer Speicherkapazität von 1,2 MByte, insgesamt also 6 MByte untergebracht (mit einem solchen Kassetten-Laufwerk von Amlyn ist in den USA übrigens auch auf Wunsch der Basis 108 zu haben).

Auswahlkriterien

In dieser Liste sind die Auswahlkriterien für Floppy-Disk- und für Plattenlaufwerke zusammengefaßt.

— Erforderliche Speicherkapazität?
— Mittlere Zugriffszeit?
— Formatierer integriert?
— Controller lieferbar? Wo untergebracht?
— Verkettung zum Anschluß mehrerer Laufwerke möglich?
— Kompatibilität erforderlich? Womit?
— Wie erfolgt die Datensicherung (Back-up)? Durch Duplizieren auf Floppy-Disk, auf Wechselplatte oder mit Magnetband (Kassette)?
— Anschaffungspreis absolut und in DM/KB?
— Bei Floppy-Disk-Laufwerken und Wechselplattenlaufwerken: Handelsübliches Speichermedium oder nur Diskette beziehungsweise Platte eines Lieferanten verwendbar?
— Daten- und Programmaustausch nötig?
— Lieferbarkeit und Service (wo und wie schnell sind Reparaturen möglich? Wird ein Wartungsvertrag angeboten und wenn ja zu welchen Kosten?)

Das Preis-/Leistungsverhältnis für Massenspeicher kann grob nach den Kosten pro Kilobyte Speicherkapazität beurteilt werden.

Beispiel für ein Floppy-Disk-Doppellaufwerk (Speicherkapazität pro Diskette 170 KByte), das 1700 Mark kostet: 1700 DM : 340 KByte = 5 DM/KByte.

Sie können natürlich auch in Pfennig/Byte oder wie die Amerikaner in Cents/Bit rechnen, handeln sich dabei aber unpraktisch viele Kommastellen ein.

Bei den Vergleichen müssen Sie
— durchgehend mit formatierten (vorzuziehen) oder unformatierten Bytes rechnen,
— bei den Investitionskosten außer dem Preis des Laufwerkes gegebenenfalls auch die Kosten des entsprechenden Controllers und der erforderlichen Software (zum Beispiel Plattenbetriebssystem) berücksichtigen.

Da im allgemeinen das System mit der größeren Kapazität auch den niedrigeren Preis pro Byte aufweist, nützt Ihnen ein Vergleich anhand der Kosten/Kilobyte nur dann etwas, wenn Sie auch wissen, wieviel Kapazität mindestens benötigt wird und wieviel Sie im Höchstfall ausgeben wollen (Überkapazität ist unschädlich). Ein Vergleich der KByte-Preise von Platten-Laufwerken einerseits und Floppy- oder Kassettenlaufwerken andererseits führt insofern etwas in die Irre, als auf ein Plattenlaufwerk mit 5-MByte-Platte eben nur 5 MByte Daten und Programme passen, auf mehreren Floppy-Disks beziehungsweise Kassetten, die man bei Bedarf einlegt, aber beliebig viele Daten und Programme gespeichert werden können.

Die wichtigsten Begriffe

In dieser Übersicht sind die Stichworte zu den Kapiteln Floppy-Disk-Laufwerke und Plattenlaufwerke zusammengefaßt.

Aufzeichnungsdichte: Bezeichnet die Anzahl der Speicherbits, die pro Zoll Spurlänge aufgezeichnet werden können. Maßeinheit: bpi (bit per inch).

Aufzeichnungsverfahren: Die Aufzeichnung erfolgt entweder mittels Frequenzmodulationsverfahren (FM) für einfache oder beispielsweise dem MFM-Verfahren (Modified Frequency Modulation) für doppelte Aufzeichnungsdichte.

bpi: Maßeinheit für die Aufzeichnungsdichte; Abkürzung steht für bit per inch (Bit pro Zoll).

bps (bit per second): Maßeinheit für die Datenübertragungsrate.

Daisy Chain → Verkettung

Datenübertragungsrate: Geschwindigkeit, mit der Daten vom Rechner zum Speicher oder umgekehrt übertragen werden können; wird in Bit/Sekunde (bps) angegeben.

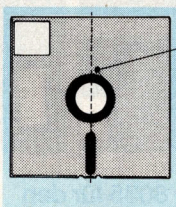
Disk: Magnetplatte; manchmal auch für Floppy-Disk verwendet.

Diskette → Floppy-Disk

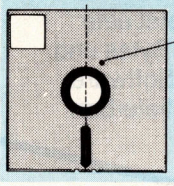
Disk-Drive → Plattenlaufwerk

DOS (Disk Operating System): Betriebssystem oder Teil davon, das für das Arbeiten mit Floppy- bzw Plattenlaufwerken erforderlich ist.

double/single density: Die Aufzeichnung kann entweder mit doppelter (double density, DD) oder aber einfacher Dichte (single density, SD) erfolgen.



Einseitig beschreibbar
Einseitig beschreibbare Disketten können an der Position der Index-Lochung erkannt werden



Doppelseitig beschreibbar
Die Index-Lochung liegt weiter von der imaginären Mittellinie entfernt

double/single-sided: Die Disketten können bei einem Teil der angebotenen Floppy-Laufwerke einseitig (single sided), bei vielen auch beidseitig (double sided) beschrieben werden.

Dual-Floppy: Doppel-Floppy-Laufwerk; Speichereinheit mit zwei Floppy-Disk-Laufwerken. Bei der Angabe der Speicherkapazität ist es zweckmäßig nachzufragen, ob sich die angegebene Kapazität pro Diskette oder für das Laufwerk insgesamt, also für zwei Disketten versteht.

Error Rate → Fehlerrate

Fehlerrate: In der Regel werden angegeben: korrigierbare Lesefehler, nicht korrigierbare Lesefehler und Suchfehler (englisch: recoverable bzw. non recoverable error rate bzw. seek error rate). Remex nennt zum Beispiel für seine Laufwerke RFD 2000/4000 ein Bit in 10^9 für korrigierbare Lesefehler, ein Bit in 10^{12} für nicht korrigierbare Lesefehler und ein Bit in 10^6 für Suchfehler.

Floppy-Disk (Diskette): Speichermedium mit 8 Zoll (Normalversion) oder 5¼ Zoll (Minifloppy) Durchmesser. Es gibt noch weitere, bislang aber nicht weitverbreitete Durchmesser.

formatierte/unformatierte Speicherkapazität: Die Daten werden auf der Floppy in Blöcken von beispielsweise 128 oder 256 Byte aufgezeichnet. Pro Block wird außerdem Platz für Lücken definierter Länge am Anfang und Ende jedes Blockes für Spur- und Sektornummern, Prüfbytes und ähnliches benötigt. Durch diese »Formatierung« der Daten können bis zu 40% Brutto-Speicherkapazität (unformatiert) verloren gehen. Nur die formatierte Speicherkapazität steht dem Anwender auch wirklich zur Speicherung seiner Programme oder Daten zur Verfügung.

Hard Disk: Magnetplatte; im Gegensatz zur Floppy-Disk nicht flexibel.

Hardsektorierung: Die einzelnen Sektoren einer Diskette werden jeweils durch eingestanzte Löcher gekennzeichnet. Diese Löcher werden optisch abgetastet. Die Hardsektorierung nutzt den verfügbaren Speicherplatz besser als die Softsektorierung.

IBM-kompatibel: Mit den von IBM verwendeten Aufzeichnungsverfahren und -formaten arbeitende Laufwerke und Datenträger. Die Kompatibilität spielt vor allem beim Datenaustausch mit Großrechnern und Rechenzentren eine Rolle.

Laufwerk: Bezeichnet zunächst nur die Antriebseinheit für Speicher. Wichtig bei der Lektüre von Prospekten und Zeitschriften: Nicht jedes Laufwerk wird im Gehäuse als aufstell- und anschlußfertige selbständige Speichereinheit angeboten.

Lebensdauer: Wird bei Laufwerken in der Regel in Betriebsstunden bzw. Jahren (z.B. 15000 Betriebsstunden oder 5 Jahre) angegeben. Zum Teil findet sich nur die MTBF (mittlere störungsfreie Zeit), meist aber beide Werte. Bei den Speichermedien wird die Lebensdauer meist in Zugriffen/Spur (englisch: passes per track) angegeben. Remex nennt z.B. bei den Laufwerken RFD 2000/4000 »$3{,}5 \times 10^6$ Zugriffe je Spur«. Der Anwender kann die Zahl der Zugriffe in der Regel nicht zählen und auch nur mühsam schätzen. Beachten Sie, daß Floppy-Disks einer zum Teil erheblichen mechanischen Abnutzung unterliegen; es ist zweckmäßig, in Zusammenarbeit mit dem Anbieter abzuschätzen, nach wieviel Wochen oder Monaten die Disketten sicherheitshalber gewechselt werden sollten. Eine Angabe, wie lange Daten auf einer Diskette zur Archivierung sicher gespeichert werden können, fehlt meistens.

Microfloppy: Der Durchmesser des Mediums (Diskette) ist — noch — nicht genormt und liegt zwischen 3 und 4 Zoll (Bild).

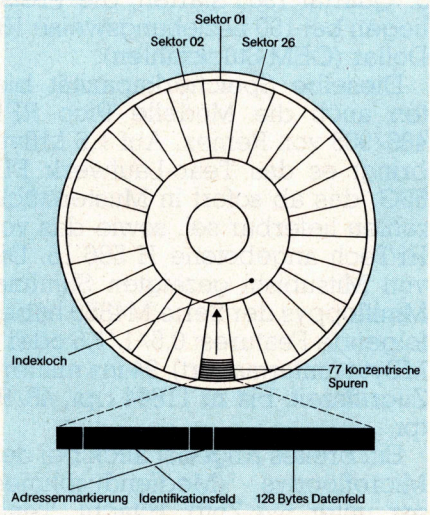
Minifloppy: Der Durchmesser des Datenträgers (Diskette) beträgt 5¼ Zoll (Bild).

Spurdichte: Gibt an, wie viele Spuren auf einem Zoll Breite nebeneinander untergebracht sind. Maßeinheit: tpi (tracks per inch).

Maßeinheit für die Spurdichte; Abkürzung steht für tracks per inch (Spuren pro Zoll).

unformatiert → formatierte/unformatierte Speicherkapazität

Winchester-Platte: Eine Platten-(Laufwerks-)konstruktion, bei der Platte und Schreib-/Lesekopf in einem hermetisch geschlossenen Gehäuse untergebracht sind.



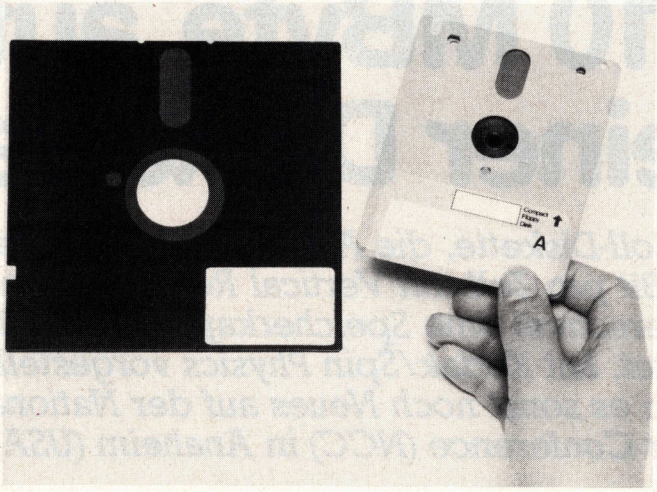
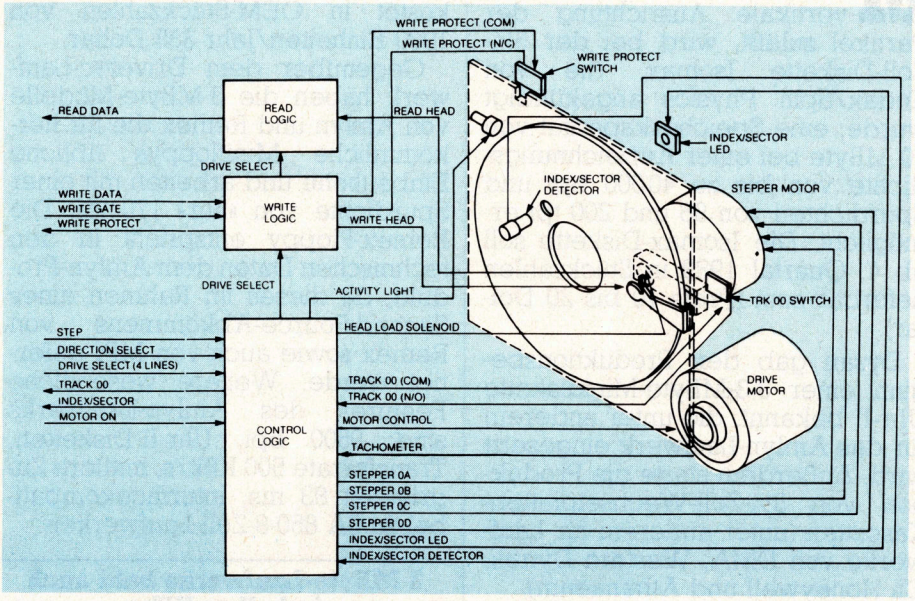


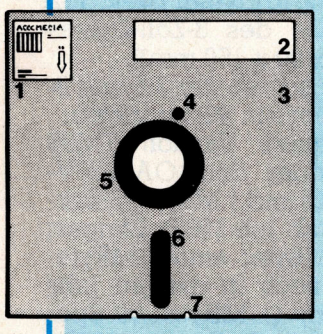
Bild links außen: Spur- und Sektorlage bei 8-Zoll-Diskette (Softsektoriert; 26 Sektoren à 128 Bytes; 77 Spuren)
Bild: BASF

Bild links: Größenvergleich zwischen 5,25 und 3 Zoll-Floppy

Zugriffszeit: Für den Anwender ist in der Regel nur die sogenannte »mittlere Zugriffszeit« interesant, das heißt die Zeit, die durchschnittlich von der Befehlsgabe bis zum Lesen eines Blockes vergeht. Sie setzt sich zusammen aus der Zeit, die erforderlich ist, um den Kopf auf die gewünschte Spur zu positionieren, der Latenzzeit, die vergeht, bis sich die Platte oder Floppy bis zum gesuchten Sektor gedreht hat und gegebenenfalls noch der Kopfladezeit, die benötigt wird, um den Kopf zum Lesen an die Diskette anzupressen. Da sich die Latenzzeit ohne weiteres aus der Umdrehungsgeschwindigkeit ergibt, wird manchmal nur die Zugriffszeit von Spur zu Spur angegeben, die deutlich niedriger liegt als die durchschnittliche Zugriffszeit.

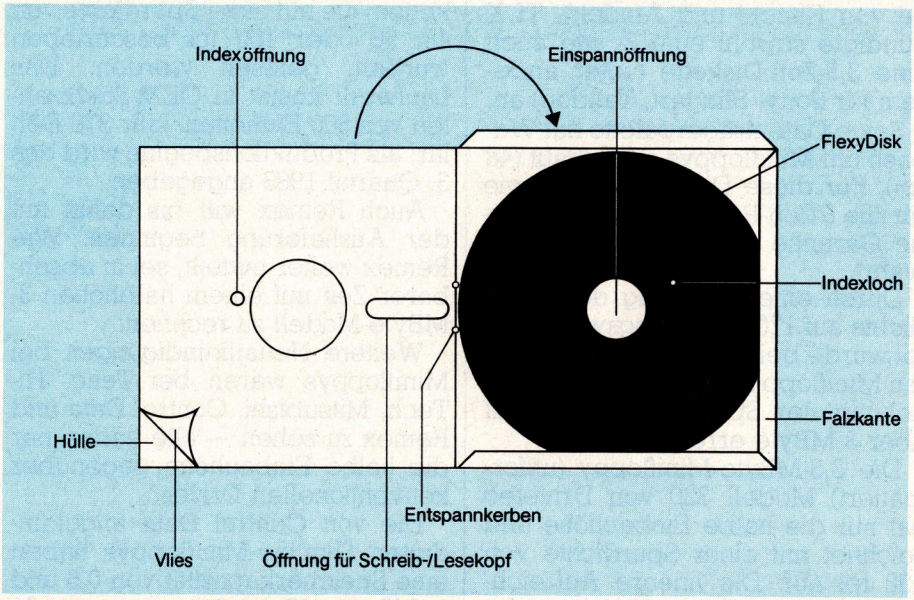


Schematische Darstellung eines Floppylaufwerks Bild: Shugart



1 Etikett
2 auswechselbares Beschriftungsschild
3 Schutzhülle
4 Index-Lochung
5 Antriebslochung
6 Lese/Schreiblochung
7 Schreibschutzvorrichtung

Die wichtigsten Details: Schematische Außenansicht einer Diskette



Aufbau einer Diskette (Schema)

10 MByte auf einer Diskette

Eine 5,25-Zoll-Diskette, die Aufzeichnungsdichten von 40000 Bit pro Zoll mit Vertical Recording und damit umgerechnet eine Speicherkapazität von 10 MByte bietet, hat Kodak/Spin Physics vorgestellt. Was gab es sonst noch Neues auf der National Computer Conference (NCC) in Anaheim (USA)?

Mittels einer Magnetschicht, die eine longitudinale und vertikale Ausrichtung der Partikel zuläßt, wird bei der 5¼-Zoll-Diskette Isomax, die von Kodak/Spin Physics angekündigt wurde, eine Speicherkapazität von 10 MByte bei einer Aufzeichnungsdichte von bis zu 40000 bpi und Spurdichten von 96 und 200 tpi ermöglicht. Die Isomax-Diskette soll ab 4. Quartal 1983 in Stückzahlen lieferbar sein (Preis 10 bis 20 Dollar).

Dysan gab den Produktionsbeginn einer 3,3-MByte-Minidiskette Uhr-II bekannt, die unter anderem für das Amlyn-Laufwerk eingesetzt wird. Außerdem starte die Produktion von 5¼-Zoll-Wechselplatten-Kassetten (unter anderem für Laufwerke von DMA, Western Dynex, Cii Honeywell und Athenaeum).

Maxell meldet, daß die 3-Zoll-Diskette nun in Stückzahlen lieferbar sei (unter anderem für Laufwerke von Hitachi und Amdek). TDK kündigte sowohl eine 3- wie auch eine 3,5-Zoll-Diskette (unter anderem für Sony, Shugart, Tandon) an.

Seine Datenträgerpalette hat Wabash um Minifloppys ausgebaut (48 tpi). Für diese Disketten wird, wie für die 8-Zoll-Produkte, ab sofort eine Garantie von sechs Jahren gewährt.

Durch eine Erhöhung der Spurdichte auf 170 beziehungsweise 192 tpi wurde bei drei zur NCC gezeigten Minifloppy-Laufwerken eine Erhöhung der Speicherkapazität auf über 3 MByte erreicht.

Die 3,3-MByte-Minifloppy (unformatiert) Modell 320 von Drivetec hat nur die halbe Einbauhöhe und zeichnet mit einer Spurdichte von 192 tpi auf. Die lineare Aufzeichnungsdichte beträgt 9908 bpi, die mittlere Zugriffszeit 160 ms. Das Modell 320 biete die Möglichkeit, auch 48-tpi-Disketten zu lesen und kostet in OEM-Stückzahlen von 1000 Einheiten/Jahr 333 Dollar.

Gegenüber dem Drivetec-Laufwerk haben die 3-MByte-Modelle von Amlyn und Remex die für herkömmliche Minifloppys übliche Einbauhöhe und arbeiten mit einer Spurdichte von »nur« 170 tpi. Die Remex-Floppy entspricht in den technischen Daten dem Amlyn-Produkt, da dieses im Rahmen eines Second-Source-Abkommens von Remex sowie auch von MPI gefertigt werde. Weitere wesentliche Features des Amlyn-Laufwerks sind: 9500 bpi, Uhr-II-Disketten, Transferrate 500 KBit/s, mittlere Zugriffszeit 83 ms, interfacekompatibel zu SA 850-8-Zoll-Laufwerken.

3 MByte-Laufwerke bald auch in halber Höhe

Es können laut Amlyn auch Disketten, die mit einer Spurdichte von 48, 96 oder 100 tpi beschrieben wurden, gelesen werden. Das Laufwerk kostet in OEM-Stückzahlen von 500 Einheiten/Jahr 370 Dollar; als Produktionsbeginn wird das 3. Quartal 1983 angegeben.

Auch Remex will bis dahin mit der Auslieferung beginnen. Wie Remex weiter mitteilt, sei in absehbarer Zeit mit einem halbhohen 3-MByte-Modell zu rechnen.

Weitere Neuankündigungen bei Minifloppys waren bei Teac, Hi-Tech, Mitsubishi, Control Data und Remex zu sehen — alle haben nur die halbe Einbauhöhe gegenüber konventionellen Drives.

Die von Control Data angekündigten Slimline-Minifloppys haben eine Speicherkapazität von 0,5 und 1 MByte (48 beziehungsweise 96 tpi). Die Stückzahllieferung soll im 3. Quartal 1983 starten, die Preise liegen bei 150 beziehungsweise 185 Dollar (OEM-Stückzahlen).

Dieselbe Speicherkapazität bieten auch die Modelle Pico RFD 485/965 von Remex. Auf 1,6 MByte bringt es das Teac-Laufwerk FD-55G, das ab sofort in Musterstückzahlen lieferbar sei, sowie das von Hi-Tech angebotene H 596-16. Die von Mitsubishi gezeigten Slimline-Minifloppys der Serie M4850 haben folgende Features: 0,5/1/1,6 oder 2 MByte (unformatiert), 94 ms mittlere Zugriffszeit, bis zu 11844 bpi, 48/96 tpi.

Ein breites Angebot auch bei den Microfloppys (Mediendurchmesser unter 5¼ Zoll): Hitachi, Teac, Sankyo und Amdek offerierten 3-Zoll-Floppys; Shugart, Tandon, Sony und Mitsubishi 3,5-Zoll-Laufwerke und IBM sein 4-Zoll-Drive.

Das 4-Zoll-Floppy-Laufwerk IBM 341 hat eine Speicherkapazität von 358 KByte (unformatiert, single sided, single density). Die Aufzeichnung erfolgt mit 6855 bpi und 68 tpi. Das IBM 341 soll vorerst nur in den USA vertrieben werden und bei sehr großen OEM-Abnahmemengen (10000 Stück/Jahr) 165 Dollar kosten. Ein Controller für das Laufwerk, dessen Interface keinem Standard entspricht, wird für 31 Dollar angeboten. Als Datenträger findet, wie bei den 3- und 3,5-Zoll-Laufwerken, eine in eine Hartplastikkassette eingebettete Diskette Verwendung. Die Abmessungen des Laufwerks betragen 66,5 mm x 119 mm x 169 mm.

Als direkter Größenvergleich seien hier die Maße des 3,5-Zoll-Laufwerks von Mitsubishi genannt (40,6 mm hoch, 101,6 mm breit, 132 mm tief) sowie des 3-Zoll-Drives von Teac (40 x 90 x 150 mm^3).

Die Leistungsdaten der 3,5-und 3-Zoll-Microfloppys in Stichpunkten:

3,5-Zoll-Medium: Tandon TM35 (minifloppy- oder Sony-OA-D30V-kompatibel, 0,5 und 1 MByte, Stückzahllieferung ab Juli 1983); Shugart SA 300 (0,5 MByte; eine 1-MByte-Version soll noch in diesem Jahr kommen); Sony (1 MByte, 80 Spuren, lieferbar ab 3. Quartal 1983); Mitsubishi (0,25 und 0,5 MByte, 80 Spuren, minifloppykompatibel, Produktionsstart: 4. Quartal 1983).

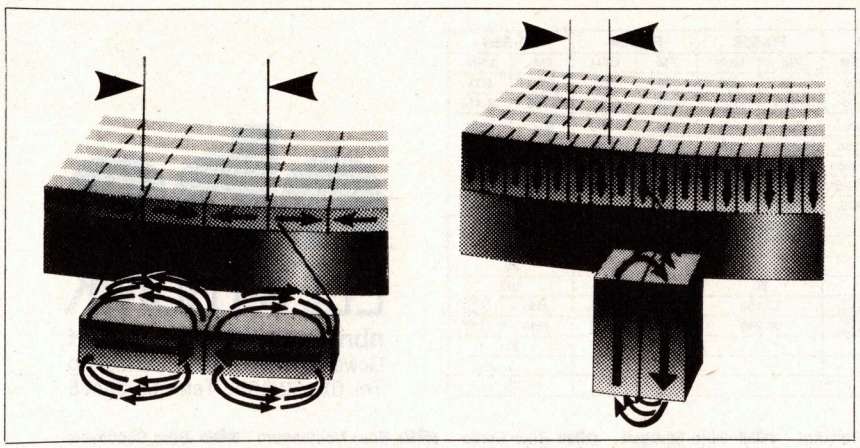
3-Zoll-Medium: Teac FD-30 (0,125 und 0,25 MByte, 40 Spuren, steckerkompatibel mit FD-50/FD-55); Hitachi (0,25/0,5 MByte); Sankyo FDU-300 (bis 0,5 MByte) und Amdek (bis 0,5 MByte). □

Stichwort: Vertical Recording

Die Speicherung von Daten auf einem magnetischen Datenträger, also beispielsweise einer Floppy-Disk, kann man etwas vereinfacht wie folgt darstellen: Jedes Bit, also eine Null oder eine Eins, wird durch einen winzig kleinen Permanent-Magneten innerhalb der magnetisierbaren Oberfläche dargestellt. Diese »Permanent-Magneten« liegen der Länge nach einer hinter dem anderen in der Spur einer Diskette. Eine Eins kann beispielsweise dadurch dargestellt werden, daß, in der Aufzeichnungsrichtung gesehen, der Nordpol vorn liegt und eine Null dadurch, daß der Südpol vorn liegt. Ein Permanent-Magnet ist im Prinzip lang und dünn. Ein bestimmtes Verhältnis von Länge zu Breite darf nicht unterschritten werden, sonst liefert er entweder kein sauberes eindeutiges Signal mehr beim Lesen oder demagnetisiert sich sogar selbst. Heute kommt man bei 1-MByte-Floppys auf Aufzeichnungsdichten von etwa 6500 Bits/Zoll bei konventioneller Technik (longitudinal recording). 10000 bis 12000 bpi lassen sich heute bei magnetischen Datenträgern erreichen, 25000 bpi gelten als die theoretische Grenze.

Ordnet man die Permanent-Magneten nun nicht wie bisher parallel zur Spur, sondern senkrecht zur magnetisierbaren Oberfläche und damit zur Spur an (eine Eins würde dann beispielsweise bedeuten, daß der Nordpol oben ist, eine Null, daß der Südpol oben ist), dann läßt sich die Aufzeichnungsdichte — die Angaben schwanken noch etwas — mindestens verzehnfachen: Man bringt also gut und gerne 100000 Bit pro Zoll und damit auf einer 5,25-Zoll-Diskette 5 bis 10 MByte unter. Das Ganze nennt sich Vertical oder Perpendicular Recording. Gearbeitet wird daran seit einigen Jahren bei verschiedenen Firmen in den USA und in Japan. Mit einer Lieferbarkeit entsprechender Disketten und Laufwerke rechnet man in den USA ab Ende 1983 beziehungsweise für 1984. Dabei konzentriert sich die Entwicklung, soweit sie nach außen sichtbar wird, derzeit in erster Linie auf die Disketten; als Laufwerk werden derzeit beispielsweise bei Vertimag konventionelle Geräte verwendet, die lediglich für die vertikale Aufzeichnung modifiziert wurden. Die Firma Vertimag, die mit der Produktion im nächsten Jahr beginnen will, rechnet damit, daß ein 5-MByte-Laufwerk etwa 500 Dollar kosten wird. Die Firma erwartet für die Zukunft, daß mit der vertikalen Aufzeichnung Aufzeichnungsdichten von 100000 x 100000 Bit pro Quadratzoll erreicht werden können — im Gegensatz von 10000 x 10000 Bit pro Quadratzoll beim Longitudinal Recording.

Das Vertical Recording bietet damit aber auch höhere Bit-Dichten, als sie bei der optischen Aufzeichnung mit Laser oder bei Halbleiterspeichern erreichbar seien.



Lage der »Permanentmagnete« in der Oberfläche bei konventioneller (links) und vertikaler (rechts) Aufzeichnung

Fortsetzung von Seite 119

Electronics für Atari 400/800, Sears Video-Spiele und den TI 99/4A von Texas Instruments feil.

Ein »Joyboard« bietet Amiga Corp. an: Das ist eine kleine Plattform, ähnlich einer Badezimmerwaage, auf die sich der Spieler stellt. Die beim Spielverlauf entsprechenden Körperbewegungen sollen von dieser Vorrichtung erfaßt und in entsprechende Steuersignale umgesetzt werden. Vielleicht eine neuer Weg, um Skigymnastik vom Computer anleiten und kontrollieren zu lassen? Oder kündigt sich hier die Video-Fahrschule der Zukunft an?

»Bio-Maus« und andere Exoten

Mancher dehnt auch den Begriff Maus seltsam weit: Edu-Ware Services propagiert die »Bio-Feedback-Mouse«. Gemeint ist die Steuerung von Computerabläufen in Abhängigkeit von Hautwiderstand oder Säuregehalt, Pulsfrequenz und ähnlichen »Biofaktoren« oder umgekehrt die Messung von Reaktionen auf Abläufe oder Informationen am Bildschirm. Kein Zweifel allerdings: Faßt man den Begriff »Benutzerschnittstelle« weit genug, bietet sich hier ein unabsehbares Reservoir an Ideen und — vielleicht — Manipulationsmöglichkeiten.

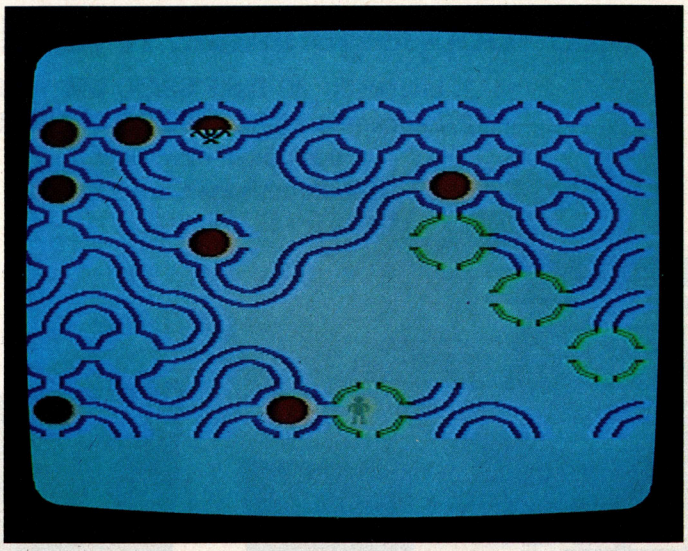
Weniger exotisch und vielleicht auch für mehr konventionelle und professionelle Anwendungen geeignet ist der Grafik Commander von Preh (Bild 6), eine leitende Kunststoff-Fläche von 115 x 95 mm², auf der ein Stift aufgesetzt wird. Die elektronisch abgetastete Position dient zur Steuerung des Cursors. Dieses Tablett ist einzeln (500 bis 600 Mark) oder an eine Tastatur angebaut zu haben (etwa 1000 Mark).

Ein ähnliches Konzept findet man sogar schon bei einem Aktentaschencomputer, dem Gavilan. Er besitzt außer dem schon traditionellen mehrzeiligen LCD-Display eine etwas kleinere Fläche ähnlichen Formates, die berührungsempfindlich ist. Hier kann durch Deuten mit dem Finger eine Auswahl zwischen verschiedenen Aktionen getroffen werden (Bild 7).

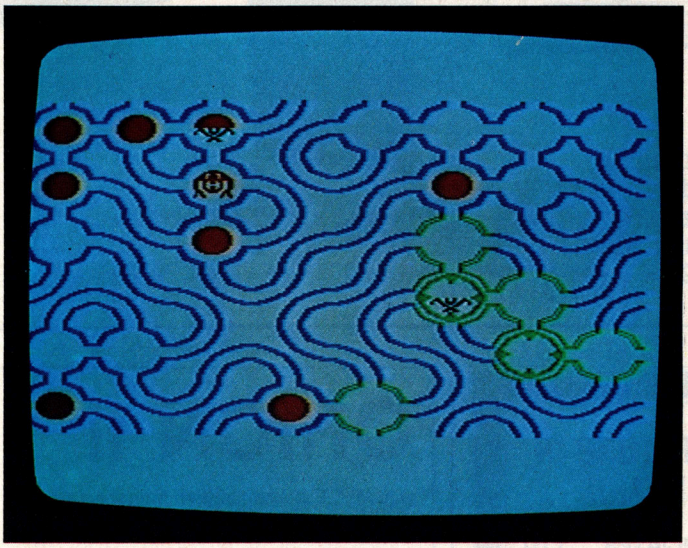
Auf jeden Fall werden die Computer freundlicher und die Mäuse sind im Kommen, mit all ihren mehr oder weniger verwandten Vettern.

(le)

Wumpus, Parsec und Alpiner: Spiele auf TI 99/4A



Wumpus-Jagd (unvollständiger Bau): Wo sitzt der Wumpus? Können Sie diese Aufgabe mittleren Schwierigkeitsgrades lösen? Preis: etwa 100 Mark



Wumpus-Jagd (vollständiger Bau): Haben Sie den Aufenthaltsort des Wumpus richtig ermittelt?

Wie alle Geräte der Preisklasse bis 1000 Mark wird der Home-Computer TI 99/4A nicht nur für ernsthafte Anwendungen eingesetzt, sondern sehr häufig auch als Spielgerät benutzt. Einige Spiele-Module werden hier vorgestellt.

Wumpus-Jagd gehört zu den Spielen, mit denen man sein logisches Denkvermögen und das Gedächtnis trainieren kann. So ganz ohne Zeitdruck kann man hier, beschaulich vor sich hingrübelnd, vor dem Bildschirm sitzen und in aller Ruhe das Versteck des gefährlichen Wumpus auskundschaften. Als mutiger Jäger ist man, mit einem einzigen Pfeil bewaffnet, in den Wumpus-Bau eingedrungen und steht nun, doch leicht zitternd, in einer Höhle des 48 Segmente großen Baus. Aus alten Überlieferungen weiß man, daß jeder Wumpus-Bau zwei Sumpflöcher enthält. Alle Höhlenwände direkt vor einem Sumpfloch sind von einem grünen Gewächs überwuchert. Vorsicht also, wenn Sie in eine grüne Höhle treten; Ihr nächster Schritt könnte Ihr letzter sein. Außerdem kennt ein erfahrener Jäger die Gattung wumpus wumpus als große »Kleckerliese« und weiß daher, daß man durch mindestens zwei mit Blut befleckte Höhlen kommen muß, bis man auf den Wumpus trifft. Durch dieses Wissen doch einigermaßen gerüstet, tappt man durch den Bau und erforscht so seinen Grundriß, soweit das ohne Gefahren in Kauf zu nehmen geht.

In der einfachsten Schwierigkeitsstufe ist das so ziemlich der ganze Bau, aber in der schwierigsten, der dritten Stufe, wo die Gänge zwischen den einzelnen Höhlen teilweise recht lang sind, können das eventuell nur ein paar Felder sein. Man hat nun sein Wissen über den Wumpus-Bau logisch einzusetzen und zu kombinieren, wie die noch unbekannten Gänge laufen könnten, wo die beiden Sumpflöcher sind und in welcher Höhle sich schließlich der Wumpus aufhält. Hat man seine Überlegungen beendet und glaubt seiner Sache sicher zu sein, so marschiert man an die betreffende Höhle heran und schießt seinen Pfeil hinein. Damit hat man entweder den Wumpus erwischt oder, falls man sich geirrt hat, gerät man selbst in die Fänge dieses gefräßigen Ungeheuers. Ist man sich über den Aufenthaltsort des Wumpus nicht im klaren, so kann man natürlich die Baubegehung fortsetzen. Das endet dann aber meist in einem Sumpfloch oder zwischen den Zähnen des Wumpus.

Neben den drei Schwierigkeitsstufen kennt dieses Spiel noch einige andere Erschwerungen. Da ist zum Beispiel die »Blind«-Variante, in der man nur die Höhle oder das Gangstück sieht, in dem beziehungsweise in der man sich befin- ►

Bildschirm-Darstellung	sehr gut
akustische Untermalung	sehr gut
Beeinflußbarkeit	ausgezeichnet
andauernde Spielermotivation	sehr gut
Gesamtbewertung	sehr gut

(Wertungsskala: ausgezeichnet — sehr gut — gut — mäßig — schlecht)

Parsec: Ein interessantes Spiel vom »Defender«-Typ. Seine Besonderheit sind gesprochene Warnungen, allerdings nur in englischer Sprache. Preis: etwa 148 Mark

det. Den restlichen Plan des Baues muß man im Kopf behalten. Ebenfalls eine bösartige Erschwernis ist die »Expreß«-Variante, in der man von Höhle zu Höhle springt, aber nichts über die verbindenden Gänge erfährt. Ganz schlimm wird es dann in der »Blind und Expreß«-Variante. Wer hier bestehen will, braucht neben einem guten Gedächtnis und logischem Denken auch eine Portion Glück.

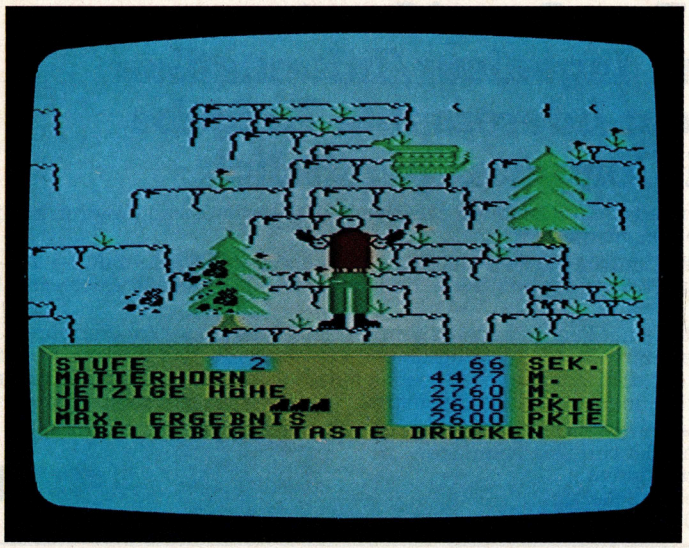
Parsec gehört zu den »Schießspielchen« vom »Defender«-Typ. Nicht weniger als sieben verschiedene Raumschifftypen gilt es abzuwehren. Manche von ihnen greifen in Gruppen an. Andere sind schwer bewaffnet und verfügen auch noch über eine automatische Zielverfolgung, das heißt, sie versuchen immer die gleiche Flughöhe anzusteuern, die man selbst gerade einhält. Da man selbst aber über eine größere Steig- und Sinkgeschwindigkeit verfügt, kann man die Angreifer durchaus ausmanövrieren. Trotzdem wird dem Piloten das Leben nicht gerade leicht gemacht, nicht zuletzt auch dadurch, daß die Laserkanone des Jägers bei Dauerfeuer schnell überhitzt, was zur Explosion des eigenen Schiffes führt. Da auch der Energievorrat des Jägers beschränkt ist, sollte man annehmen, daß sparsamer Umgang mit der »Energie-Munition« belohnt wird. Das ist aber leider nicht so. Man bekommt für jeden erfolgreich durchgeführten Tankvorgang Zusatzpunkte zugeteilt. Für Punktejäger heißt das also, sich zwar vor Laserüberhitzung zu hüten, aber ansonsten feuern, was das Zeug hält, ganz egal, ob ein Ziel in Sicht ist oder nicht, nur um möglichst oft nachtanken zu müssen. Etwas unlogisch, finden Sie nicht? Trotzdem macht das Spiel mit seinen vielen aufsteigenden Schwierigkeitsgraden großen Spaß und wird über lange Zeit gewiß gerne gespielt werden. Wer über einen Speech Syntheziser verfügt, der kann in diesem Spiel bereits eine brauchbare Sprachausgabe — allerdings in Englisch — erleben.

Bildschirm-Darstellung	ausgezeichnet
akustische Untermalung	ausgezeichnet
Beeinflußbarkeit	sehr gut
andauernde Spielermotivation	ausgezeichnet
Gesamtbewertung:	ausgezeichnet

Gesamtbewertung: ausgezeichnet — sehr gut — gut — mäßig — schlecht)

Alpiner läßt den Spieler als Bergsteiger einige der schwierigsten Berge unserer Erde erklimmen. Dabei behindern ihn die überall herumstehenden und -liegenden Tiere wie Steinböcke, Bären, Berglöwen und so weiter. Natürlich muß man einige Hindernisse einbauen, um ein Spiel abwechslungsreich und spannend zu machen. Dazu genügen unbewegliche Hindernisse wie Bäume und Baumstümpfe und sich bewegende wie Steinschlag, Lawinen und ähnliches. Gegen die oben erwähnten Tiere gibt es nur wenig einzuwenden; man fragt sich jedoch zum Beispiel warum man sie als sich nicht bewegende Hindernisse konzipiert hat. Der wandernde Schneemensch mag als Gag durchaus angebracht sein, was aber haben Stinktiere und Klapperschlangen am Matterhorn zu suchen? Obwohl man das friedliche Thema — vom Absturz des Kletterers einmal abgesehen — eigentlich begrüßen müßte, bringt dieses Spiel nicht sonderlich viel. Die Sprachausgabe — nur für Speech Synthesizer-Besitzer — wird hier allerdings bereits sinnvoll eingesetzt. Die gesprochenen Warnungen verkürzen die Reaktionszeiten erheblich. Für den an »Action« interessierten Spieler ist dieses Modul mit Sicherheit kein Dauerbrenner, vielleicht ist es aber der Startschuß zum sinnvollen Einsatz des Speech Synthezisers auf dem Telespiel-Sektor.

(Joseph Weigand)



Alpiner: Was sucht die Klapperschlange am Matterhorn? Preis: ebenfalls um die 150 Mark

Bildschirm-Darstellung	gut
akustische Untermalung	ausgezeichnet
Beeinflußbarkeit	sehr gut
andauernde Spielermotivation	gut
Gesamtbewertung	gut

(Wertungsskala: ausgezeichnet — sehr gut — gut — mäßig — schlecht)

Fortsetzung von Seite 49

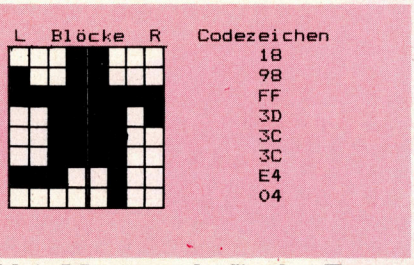


Bild 1. Mustercode für das Tanzmännchen

nen ersten Zugang zum erweiterten Basic, sondern wird darüber hinaus dem bereits geübten Benutzer zu einem übersichtlichen Nachschlagewerk.

Für jeden Befehl wird zuerst das Format angegeben. Dann wird er ausführlich beschrieben und an einem Beispiel erläutert. In den meisten Fällen handelt es sich bei dem Beispiel um ein vollständiges Programm.

```
Das Programm endet mit <FCTN-4> (CLEAR).

        100 REM ***** TANZMAENNCHEN *****
        110 CALL CLEAR
        120 CALL CHAR(96,"1898FF3D3C3CE404").
        130 CALL CHAR(97,"1819FFBC3C3C2720")
        140 CALL VCHAR(12,16,96)
        150 FOR VERZOEG=1 TO 50 :: NEXT VERZOEG
        160 CALL VCHAR(12,16,97)
        170 FOR VERZOEG=1 TO 50 :: NEXT VERZOEG
        180 GOTO 140
```

Bild 2. Programm »Tanzmännchen«

Eine interessante Anwendungsmöglichkeit, mit der Programme lebhafter gestaltet werden können, bietet das Unterprogramm CHAR. Ein Auszug aus dem Buch in Bild 1 zeigt, wie ein Tanzmännchen definiert wird. Mit dem Programm in Bild 2 wird dieses Männchen zum Tanzen gebracht. »Sprites« sind Grafiksymbole, die mit unterschiedlicher Geschwindigkeit in alle Richtungen in Bewegung versetzt werden können und aus Weltraumspielen wohlbekannt sein dürften. Das Unterprogramm Sprite wird ebenfalls genauestens beschrieben. Mehrere Unterprogramme beeinflussen den Zustand der Sprites oder informieren über sie.

Das Buch ist jedem zu empfehlen, der in TI Extended Basic programmiert und die Fähigkeiten dieser Sprache voll ausschöpfen möchte.

(Gu)

Info: TI Basic/Extended Basic für Anfänger und Fortgeschrittene, Texas Instruments, Freising April 83, ISBN 3-88078-039-0

Inserenten in dieser Ausgabe

aaa	77
Abacomp	36
Abor	87
Arhelger	36
Basic Bär	47
Bense	105
bfi	25
Bosch Computertechnik	151
BSP Krug	81
Büro-Elektronik-Steins	117
C.V.B. Buerschaper	79
Canon	45
cc computer studio	71
CE, Computer Elektronik	71
City Elektronik	160
Commodore	25, 27
Computer Buchladen	120-123
Computer Forum	69
Computer Studio Braunschweig	25
Corvallis	151
CS Computer Software	86
CSC	57
Cumana	143
Data Becker	43
Delta/B.A.R.	117
Digital Equipment	18/19
Dipa	5, 69
DSB	151
Frölje Elektronik	117
Füssner	47
Fujitsu	2
Graumann	145
Hofacker	59
Höfel	69
Hübner + Worm	75
IWT	146
Kohlruss	145
L.U.D.O.P.	40
Ley	75
Lifeboat Associates	119
M&T Software Verlag	50/51, 107
Maxell	145
Microdex	47
mm electronic	146
Morrow Design	53
MSB	79
Namal	151
nbn	149
NEC	36
Neumüller	117
Omecon	75
PAV	75
PTM	39
Radix	105
Reis	47
REX EDV	71
Rosenthal	63
Rekers	146
Sanyo Video	79
SE	28/29
SM	145
SoftConstruct	105
Springmann	105
Stemmos	83
Studiengemeinschaft Darmstadt	71
Texas Instruments	23
time-soft	36, 79
Toshiba	33
Trommeschläger	17

Impressum

Herausgeber: C.F. von Quadt, Otmar Weber

Chefredakteur: Michael M. Pauly (py)
Stellv. Chefredakteur: Michael Scharfenberger (sc)
Redakteure: Silvia Gutschmidt (gu), Günter Jürgensmeier (jü), Johannes Leckebusch (le), Manfred S. Schmidt (mss)
Redaktionsassistenz: Dagmar Zednik (237)
Layout: Alexander Gerhardt
Fotografie: Janos Feitser, Titelfoto: Alex Kempkens

Auslandsrepräsentation: Schweiz: Markt&Technik-Agentur, Postfach, CH-8033 Zürich, Tel. 01-362 1320

Manuskripteinsendungen: Manuskripte werden gerne von der Redaktion angenommen. Honorare nach Vereinbarung. Die Zustimmung des Verfassers zum Abdruck wird vorausgesetzt. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

Herstellung: Klaus Buck (180), Leo Eder (181)

Anzeigenleitung: Peter Schrödel (156); Anzeigenverkauf: Kurt Ebner (211), Waltraud Werner (236), Einkaufsführer: Marion Heinrichs (118)

Anzeigenverwaltung und Disposition: Patricia Schiede (172)

Anzeigenformate: 1/1-Seite ist 266 Millimeter hoch und 185 Millimeter breit (3 Spalten à 58 mm oder 4 Spalten à 43 Millimeter). Vollformat 297x210 Millimeter. Beilagen und Beihefter siehe Anzeigenpreisliste.

Anzeigenpreise: Es gilt die Anzeigenpreisliste Nr. 2 vom 1. Januar 1983.
1/1 Seite sw: DM 3960,-. Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala DM 720,-. Vierfarbzuschlag DM 1860,-.
Mindestgröße Produktanzeigen = 1/8 Seite = DM 495,-. Kleinanzeigen im CP-Markt: je Druckzeile und Spalte (Spaltenbreite 43 mm) gewerblich DM 7,- (+ MwSt.), privat DM 5,- (inkl. MwSt.).
CP-Einkaufsführer: Mindesthöhe 15 mm, Mindestabnahme 6 Eintragungen; Spaltenbreite 58 mm, je mm Abdruckhöhe DM 6,-. Chiffregebühr: DM 10,- (Inland), die gesetzliche MwSt. wird jeweils zugerechnet.

Vertriebsleitung, Werbung: Hans Hörl

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Verlagsunion Zeitschriften-Vertriebs-KG, Friedrich-Bergius-Str. 7, 6200 Wiesbaden, Telefon 06121/266-0.

Erscheinungsweise: »Computer persönlich« erscheint 14täglich, Mittwoch.

Bezugsmöglichkeiten: Abonnenten-Service: Silvia Augustin Tel. 089/4613238. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen. Das Abonnement verlängert sich um ein Jahr, wenn es nicht zwei Monate vor Ablauf gekündigt wird.

Bezugspreise: Das Einzelheft kostet DM 4,-. Der Abonnementspreis beträgt im Inland DM 78,- pro Jahr für 26 Ausgaben. Darin enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren. Der Abonnementspreis erhöht sich um DM 25,- für die Zustellung im Ausland, für die Luftpostzustellung in Ländergruppe 1 (z.B. USA) um DM 65,-, in Ländergruppe 2 (z.B. Hongkong) um DM 100,-, in Ländergruppe 3 (z.B. Australien) um DM 130,-.

Druck: R. Oldenbourg, Hürderstr. 4, 8011 Kirchheim
Auch Anschrift für Beihefter und Beilagen.

Urheberrecht: Alle in »Computer persönlich« erschienenen Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten. Reproduktionen gleich welcher Art, ob Fotokopie, Mikrofilm oder Erfassung in Datenverarbeitungsanlagen, nur mit schriftlicher Genehmigung des Verlages. Anfragen sind an Hans Hörl zu richten. Für Schaltungen und Programme, die als Beispiele veröffentlicht werden, können wir weder Gewähr noch irgendwelche Haftung übernehmen. Aus der Veröffentlichung kann nicht geschlossen werden, daß die beschriebenen Lösungen oder verwendeten Bezeichnungen frei von gewerblichen Schutzrechten sind.

© 1983 Markt & Technik Verlagsgesellschaft mbH, Redaktion »Computer persönlich«.

Verantwortlich: Für redaktionellen Teil: Michael M. Pauly. Für Anzeigen: Peter Schrödel.

Geschäftsführer: C.F. v. Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt&Technik Verlagsgesellschaft mbH, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon 089/4613-0, Telex 5-22052

ISSN 0722-0987

Telefon-Durchwahl im Verlag:

Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089-4613 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.

Info CP in ***Düsseldorf*** und ***Berlin***
☎ ★ 15982 ↓ (Bildschirmtext)

»Screen Inverse«

Wer seinen TI 99/4A an einem monochromen Datenmonitor betreibt, den stört sicher der helle Hintergrund mit den schwarzen Zeichen. Der TI 99/4A mit Extended-Basic-Modul bietet nun eine Möglichkeit, aus der systemeigenen Interrupt-Routine in ein Anwender-Maschinenprogramm zu verzweigen, indem man in die Adresse 83C4 die Verzweigungsadresse schreibt.

Diese Möglichkeit nutzt nun das kleine Maschinenprogramm in Bild 1 aus, um die Hintergrundfarbe auf Transparent und die Zeichenfarbe auf Weiß umzustellen. Das Programm stört weder die Farbbefehle im Basic-Programm, noch die I/O-Operationen. Wer den Editor-Assembler hat, kann das Programm in seinen Editor eingeben und neu assemblieren. Ohne diese Möglichkeit bleibt nichts anderes übrig als den Tagget-Object-Code (Bild 2) in Form eines DISPLAY/FIXED 80-Datenfile auf Kassette oder Diskette abzulegen. Dann gibt man »CALL INIT :: CALL LOAD ("FILENAME. PROGRAMMNAME") :: CALL LINK ("INVRUN")« ein. Das Programm wird geladen und gestartet. Die Schirmfarben werden sich ändern, und der TI kann wie normal benutzt werden.

Wer den Assembler hat, kann auch die Farben in EQU COLOR für die Zeichen und in EQU COLMOD für den Hintergrund ändern. Im TI-Basic muß auch EQU COLTAB auf die passende Farbtabellen-Adresse abgeändert werden.

Um zum Beispiel das Screen-Inverse-Programm ohne Editor/Assembler oder Minimemory auf einen Datenträger abzulegen, kann das Programm »Tagget-Code-Editor« (Bild 3) eine gute Unterstützung sein. Im Menü werden einige Routinen angeboten. Mit »E« springt das Programm in den eigentlichen Editor. Jede Bewegung des Cursors, ob rauf oder runter, darf nur mit den Pfeiltasten vorgenommen werden. Bei Betätigung der Enter-Taste springt das Programm wieder ins Menü. Zu sehen ist beim Editieren immer nur eine Zeile mit nachgestellter Zeilennummer. Diese dient nicht nur zur Unterstützung des Schreibers, sondern wird später benötigt. Mit Hilfe der Pfeiltasten kommt man auch in eine vorhergehende Zeile, indem man versucht, den Cursor in die Bildschirmzeile Null zu bewegen und umgekehrt in eine Zeile mit höherer Nummer, wenn man den Cursor zur Bildschirmzeile Vier bewegt. (L)aden und (S)peichern benutzt man für ein Kassetten- und Diskettenge-

```
*****************************************
*       DIESES PROGRAMM ERLAUBT ES DEN
*       TI 99/4A MIT JEDER BELIEBIGEN
*       BILDSCHIRMFARBE ZU BETREIBEN
*****************************************
        TEXT 'WRITTEN BY M.FRANZ 03.04.1983'
        EVEN
USRWP   BSS  32
COLMOD  EQU  >87F0          VDP R7 HINTERGRUNDFARBE KLAR
RUNFLG  EQU  >8344          BASIC RUNFLAG
VDPWD   EQU  >8C00          VDP WRITE DATEN ADRESSE
VDPWA   EQU  >8C02          VDP WRITE ADRESSEN ADRESSE
BASADR  EQU  >0070          TI 99/4 BASIS ADRESSE
COLTAB  EQU  >480E          VDP EXT. BASIC COLOR TABLE ADRESSE
COUNT   EQU  >0008          COLOR COUNT
COLOR   EQU  >F0F0          ZEICHEN FARBE
USRADR  EQU  >83C4          TI 99/4A USER ADRESSE
GPLWP   EQU  >83E0          TI 99/4 WP ADRESSE
STATUS  EQU  >837C
*
        DEF  INVRUN
INVRUN  LWPI USRWP
        LI   R0,RUN
        MOV  R0,$USRADR     LADE INVERSE ADRESSE
        CLR  R0
        MOVB R0,$STATUS     LOESCHE STATUS
        LWPI GPLWP
        B    $BASADR        RETURN TO BASIC
*
RUN     LWPI USRWP
        CLR  R0
        MOVB $RUNFLG,R0     TEST AUF PROGRAMM-MODUS
        JNE  EXIT
*
        LI   R0,VDPWA       LADE VDP WRITE ADRESSE
        LI   R2,COLMOD      LADE HINTERGRUNDFARBE
        BL   $WRITE         JUMP WRITE
*
        LI   R2,COLTAB      LADE COLOR TABELLEN ADRESSE
        BL   $WRITE         JUMP WRITE
*
        LI   R3,COUNT       LADE ZAEHLER
        LI   R2,COLOR       LADE COLOR
        LI   R0,VDPWD       LADE VDP WRITE DATEN
LOOP    BL   $WRITE         JUMP WRITE
        DEC  R3             ZAEHLER -1
        JNE  LOOP           SCHLEIFE
*
EXIT    LWPI GPLWP          LADE TI WP
        RT                  RETURN TO BASIC
*
**** UNTERPROGRAMM SCHREIBE ZUM VDP ***
*
WRITE   SWPB R2             TAUSCHE BYTE
        MOVB R2,*R0         BYTE TO VDP
        SWPB R2             TAUSCHE BYTE
        MOVB R2,*R0         BYTE TO VDP
        RT                  RETURN
*
        END
```

Bild 1. Listing. Das Programm erlaubt die Umschaltung der Bildschirmfarbe.

electronic gmbh

NEUE PREISE # 10

Sinclair ZX81 fertig	154.—
64K-RAM-Speichererweiterung im Gehäuse	239.—
Sinclair-Drucker	198.—
Apple-Bus Computerplatinen + Peripherie	
Motherboard durchkontaktiert mit Lötstoplack	97.—
M-board 48K, gesockelt, voll bestückt + geprüft	695.—
Netzteil geprüft + kurzschlußfest	165.—
Getaktetes Netzteil geprüft + kurzschlußfest	199.—
M-board 48K + Netzteil (fertig + geprüft)	828.—
M-board 48K + Get.-Netzteil (fertig + geprüft)	858.—
M-board 48K + Netzteil + Tastatur (fertig + geprüft)	938.—
M-b 48K + Get.-Netzteil + Tastatur (fertig + geprüft)	958.—
Apollo Leergehäuse in Plastik	155.—
16K-Speicher-Karte Leerplatine	45.—
16K-Speicher-Karte geprüft	155.—
Z80-CPU-Karte Leerplatine	45.—
Z80-CPU-Karte geprüft	175.—
Controller DOS 3.3-Karte Leerplatine	69.—
Controller DOS 3.3-Karte geprüft	155.—
Parallel-Drucker-Karte Leerplatine	69.—
Parallel-Drucker-Karte geprüft	185.—
Seriell-Interface-Karte V.24 Leerplatine	45.—
Seriell-Interface-Karte V.24 geprüft	195.—
80 Zeichen-/24 Zeilen-Karte-Leerplatine	45.—
80 Zeichen-/24 Zeilen-Karte geprüft	245.—
80 Z-/24 Z-Karte + Softswitch-Schalter Leerplatine	69.—
80 Z-/24 Z-Karte + Softswitch-Schalter geprüft	295.—
PAL-Modulator/Color-Karte Leerplatine	45.—
PAL-Modulator/Color-Karte geprüft	255.—
256K-Speicher-Leerkarte + 4 Sp.ICs + Software	385.—
256K-Speicher-Karte + Software geprüft	1375.—
Ab 5 Leerplatinen (gemischt) minus 10% Rabatt !!!	
Ab 20 Leerplatinen (gemischt) minus 20% Rabatt !!!	
Disk II + Controller + Kabel DOS 3.3	879.—
Disk II (2. Laufwerk)	729.—
Disk IIF + Controller + Kabel DOS 3.3 (½ Höhe)	828.—
Disk IIF (2. Laufwerk) (½ Höhe)	678.—
UHF/TV-Modulator Universell	34.—
APOLLO II (48K-V) + UHF-Modulator fertig — Gerät + gepr.	998.—
APOLLO II (64K-V) + UHF-Modulator	1128.—
APOLLO II (64K-V) + Disk + Contr. + 12"-Monitor	2228.—
APOLLO IIA (48K-V) + Gr./Kl. + 10er Tastatur	1245.—
APOLLO IIA (64K-V) + Gr./Kl. + 10er Tastatur	1375.—
APOLLO IIA (64K-V) + Disk + Contr. + 12"-Monitor	2455.—
APPLE II + Euro (48K-V) + UHF-Modulator	2379.—
APPLE II + Euro (64K-V) + UHF-Modulator	2519.—
APPLE-DISK II Laufwerk + Contr. + Kab. DOS 3.3	1195.—
APPLE-DISK II (2. Laufwerk)	895.—
dt. Groß/Klein-Tastatur-Umbausatz	245.—
APPLESOFT Handbuch deutsch	39.—
APPLE-Integer-Basic Handbuch deutsch	44.—
APPLE-Tutorial Handbuch deutsch	38.—
APPLE-Reference Handbuch deutsch	55.—
APPLE-DOS 3.3 Handbuch deutsch	45.—
APPLE Pascal Reference + Operating Manual orig. engl.	165.—
APPLE II Pascal Handbuch deutsch (te-wi)	55.—
APPLE-II Anwenderhandbuch deutsch (te-wi)	49.—
6502-Programmier Handbuch deutsch	45.—
6502-Hardware Handbuch deutsch	39.—
6502-Anwendungen Handbuch deutsch (Rod. Zaks)	38.—
CP/M-Handbuch mit MP/M deutsch	44.—
CP/M und WordStar Handbuch deutsch	29.80
APPLE IIe (64K-V)	2995.—
APPLE IIe (64K-V) + Disk + Contr. + Kab. DOS 3.3	3845.—
APPLE IIe (64K-V) + Disk + Contr. + 12"-Monitor	4095.—
80 Z/24 Z + 64K-RAM-Karte geprüft	575.—
BASIS 108 (128K-V) 6502 + Z80-CPU	4995.—
BASIS 108 (128K-V) + Disk II + Contr. DOS 3.3	5795.—
BASIS 108 (128K-V) + Disk II + Contr. + 12"-Monitor	5995.—
BASIS 108 (128K-V) + 2★Disk II + Contr. DOS 3.3	6495.—
BASIS 108 (128K-V) + 2★Disk II + Contr. + 12"-Mon.	6645.—
TEXAS TI 99/4A	485.—
Modul-Box mit 6 Einschüben	659.—
RS232 + 8 Bit Interface-Karte	435.—
Disk-Controller-Karte	659.—
Disketten-Laufwerk	1070.—
32K-RAM-Speichererweiterungs-Modul	825.—
Extended-BASIC-Modul	329.—
Schachmeister-Modul	129.—
Statistik-Modul	165.—
SHARP PC-1210 inkl. dt. Hb.	175.—
SHARP PC-1211 inkl. dt. Hb.	265.—
SHARP PC-1212 inkl. dt. Hb.	175.—
SHARP PC-1245 inkl. dt. Hb.	165.—
SHARP PC-1251 inkl. dt. Hb.	285.—
SHARP CE-122 Drucker + Kassetten-Interface	194.—
SHARP CE-125 Drucker + Kassettenrekorder	319.—
SHARP PC-1500 inkl. dt. Hb.	459.—
SHARP CE-150 4-Farben-Drucker/Plotter + Kass.-Int.	385.—
SHARP CE-158 RS232 + 8 Bit Interface	389.—
VC 20 (5K-V)	395.—
VC 64 (64K-V)	698.—
VC 1530 Datasette	149.—
VC 1515 Matrix-Drucker	795.—
VC 1541 Floppylaufwerk	765.—
cbm 8032 SK (bewegliche Tastatur)	2195.—
cbm/PET TV-Interface (f. 9" 2.3 + 40xx Serie)	245.—
AIM 10er Pack Thermopapier	29.50
Colour-Genie EG 2000	595.—
Genie I (64K-V/Mod. 83) + 12"-Monitor	1248.—
Genie I (64K-V/Mod. 83) + Laufwerk + 12"-Monitor	2598.—
Genie II (64K-V/Mod. 83) + 12"-Monitor	1348.—
Sharp MZ-80A (48K-V)	1995.—
EXIDY SORCERER (48K-V)	1795.—
Seikosha GP-80, 8 Bit/parallel	655.—
Seikosha GP-100, 8 Bit/parallel	695.—
Seikosha GP-250, 8 Bit/parallel	895.—
Seikosha GP-700A, 8 Bit/parallel, 7 Farben	1375.—
STAR DP 510 8 Bit/parallel	1075.—
EPSON TX 80T, 8 Bit/parallel	895.—
EPSON RX 80, 8 Bit/parallel	1145.—
EPSON FX 80, 8 Bit/parallel	1675.—
APPLE/EPSON-Drucker Interface + Kabel	195.—
APPLE/EPSON-Drucker Graphik-Interface + Kabel	325.—
cbm/EPSON-Drucker Interface + Kabel	350.—
EPSON MX 80/82 Spezialdruckkopf	149.—
MX 80/82, FX/RX 80 Spezialfarbband-Kassette	15.95
MX 100 Spezialersatzfarbband	13.95
Olivetti Praxis 35 + Centronics Interface	1435.—
12"-Videb-Monitor, grün, 15 MHz Zenith ZVM-121	257.—
SANYO DD 8112, 12", gr., 18 MHz	475.—
SANYO DD 8212, 12", gr., 18 MHz	485.—
Antireflex-Universal-Folie	11.95
3M-Scotch Disketten-verstärkt in Box, I. Wahl	
5¼" 1D-Disk-softsektoriert 10er Pack	69.—
5¼" 1D-Disk-softsektoriert 100er Pack	645.—
5¼" 2D-Disk-softsektoriert 10er Pack	105.50
5¼" 2D-Disk-softsektoriert 100er Pack	969.—
3M-Scotch 5¼" Reinigungsset (2 Disketten)	67.95
BASF-Disketten in Box + Aufkleber 1. Wahl	
5¼", 0.8" soft-o. hardsektoriert 10er Pack	55.—
5¼", 0.8" soft-o. hardsektoriert 100er Pack	539.—
5¼", 0.8" soft-o. hardsektoriert 1000er Pack	5250.—
5¼", Soft-verstärkt 10er Pack	63.—
5¼", Soft-verstärkt 100er Pack	605.—
5¼", Soft-verstärkt 1000er Pack	5950.—
5¼", 0.8" 1D-softsektoriert 10er Pack	58.—
5¼", 0.8" 1D-softsektoriert 100er Pack	565.—
5¼", 0.8" 1D-softsektoriert 1000er Pack	5495.—
5¼", 0.8" 2D-softsektoriert 10er Pack	89.—
5¼", 0.8" 2D-softsektoriert 100er Pack	869.—
5¼", 0.8" 2D-softsektoriert 1000er Pack	8350.—
5¼", 1/96- softsektoriert 10er Pack	84.95
5¼", 1/96-softsektoriert 100er Pack	799.—
5¼", 2/96-softsektoriert 10er Pack	105.95
5¼", 2/96-softsektoriert 100er Pack	999.—
5¼" Reinigungs-Diskette	59.95
5¼" Disketten-Archivbox f. 10 Disks	7.50
5¼" Disketten-Archivbox 10er-Pack	72.—
8" Disketten-Archivbox f. 10 Disks	9.50
8" Disketten-Archivbox 10er-Pack	89.—
2000 Bl. Tabellierpapier (240★12" einfach) weiß oder grün/weiß perforiert	47.—
4000 Etiketten-Aufkleber (107★36 mm Doppelreihe) auf 240 mm★12" perf. Trägerpapier	66.50
16000 Etiketten-Aufkleber w. o.	249.—
10000 Etiketten-Aufkleber 1reih. (125★12")	179.—
EPROM-Löschlampe E-27 universell	79.95
Reinigungs- und Entmagnetisierungs-Kassette	10.95
OSZILLOSCOPE HAMEG ab Lager (s.V.r.)	

Bei Vorauszahlung frei Empfangsstation in der BRD! ausgenommen Papier und Etiketten, sonst NN + V.S.

electronic gmbh

Telex: 0772642 aaa-d
Habsburger Straße 134
7800 FREIBURG, Tel. (0761) 27 68 64
Bauelemente – Bausätze – µP's
Meßgeräte – Zubehör – Fachliteratur
Fachgeschäft für Elektronik · Mikrocomputer

```
00092INVERSE A0000B5752B4954B5445B4E20B4259B204DB2E46B5241B4E5A7F19FF    0001
A0012B2036B2E33B2E31B3938B3300A001CA003CB02E0C001CB0200C00507F2F3F     0002
A0044BC800B83C4B02E0B83E0B0460B0070B02E0C001CB0200B8C02B02017F2D7F     0003
A005AC0088B0202B87F0B0691B0202BE001BC802B83D4B0202B080EB06917F2C7F     0004
A0070B0203B0008B0202BF0F0B0200B8C00B0691B0603B16FDB02E0B83E07F2D0F     0005
A0086B045BB06C2BD402B06C2BD402B045B7F80EF                              0006
5003CINVRUN7FCDCF                                                      0007
:        99/4 AS     NVERSE *                                          0008
```

Bild 2. Der Tagget-Object-Code besteht aus acht Zeilen mit jeweils 80 ASCII-Zeichen, die als DISPLAY/FIXED 80-Datenfile abgespeichert oder in den Tagget-Editor eingegeben werden müssen.

```
100 ! TAGGET EDITOR REV. 1.0
110 ! WRITTEN BY M.FRANZ
120 ! HALSTENBEK 04.08.1983
130 ON ERROR 780
140 LENGTH=80 :: LENGTH2=20
150 OPTION BASE 1 :: DIM TEXT$(400)
160 JUMP$="ELNFSAZ"
170 DISPLAY AT(1,1)ERASE ALL BEEP:"     TAGGET-EDITOR 8/83":RPT$("-",28)
180 DISPLAY AT(4,1):"(E)INGEBEN":"(L)ADEN":"(N)EU":"(F)ILENAME":"(S)PEICHERN":"(
A)UFWIEDERSEHEN":"(Z)EICHENFORMAT"
190 DISPLAY AT(12,1):"FILENAME:":FILE$
200 DISPLAY AT(20,1):"LETZTE ZEILE:";HILINE
210 DISPLAY AT(15,1):"ZEICHENFORMAT:":STR$(LENGTH)
220 CALL HCHAR(3+POSITION,1,32)
230 CALL KEY(0,K,S):: IF S<=0 THEN 230
240 POSITION=POS(JUMP$,CHR$(K),1)
250 IF POSITION=0 THEN CALL SOUND(300,220,0):: GOTO 220
260 CALL SOUND(80,1400,5)
270 CALL HCHAR(3+POSITION,1,30)
280 ON POSITION GOTO 380,290,690,750,610,880,910
290 !LADEN
300 IF SEG$(FILE$,1,2)="CS" THEN FLAG=1 :: CALL CLEAR ELSE FLAG=0
310 OPEN #1:FILE$,INPUT ,DISPLAY ,FIXED LENGTH
320 FOR I=1 TO 100
330 LINPUT #1:TEXT$(I)
340 IF POS(TEXT$(I),":",1)THEN 360
350 NEXT I
360 CLOSE #1
370 HILINE=I
380 !EINGEBEN
390 I=1
400 HILINE=MAX(HILINE,I)
410 GOSUB 560 !DISPLAY LINE
420 ACCEPT AT(1,1)SIZE(-28):TEXT1$
430 CALL KEY(0,K,S)
440 TEXT$(I)=TEXT1$&RPT$(" ",28-LEN(TEXT1$))&SEG$(TEXT$(I),29,LENGTH)
450 IF K=13 THEN 550 ELSE IF K=11 AND I>1 THEN I=I-1 :: GOSUB 560 :: GOTO 500 EL
SE IF K=11 THEN 420
460 ACCEPT AT(2,1)SIZE(-28):TEXT2$
470 CALL KEY(0,K,S)
480 TEXT$(I)=SEG$(TEXT$(I),1,28)&TEXT2$&RPT$(" ",28-LEN(TEXT2$))&SEG$(TEXT$(I),5
7,LENGTH)
490 IF K=11 THEN 420 ELSE IF K=13 THEN 550
500 ACCEPT AT(3,1)SIZE(-LENGTH2):TEXT3$
510 CALL KEY(0,K,S)
520 TEXT$(I)=SEG$(TEXT$(I),1,56)&TEXT3$&RPT$(" ",LENGTH2-LEN(TEXT3$))&LINE$
530 IF K=11 THEN 460 ELSE IF K=13 THEN 550
540 IF I<100 THEN I=I+1 :: GOTO 400 ELSE 550
550 GOTO 170
560 !SUB DISPLAY
570 LINE$=SEG$("0000",1,4-LEN(STR$(I)))&STR$(I)
580 IF TEXT$(I)="" THEN TEXT$(I)=RPT$(" ",LENGTH-4)&LINE$
590 DISPLAY AT(1,1)ERASE ALL:TEXT$(I):RPT$("-",28)
600 RETURN
610 !SAVE
620 IF SEG$(FILE$,1,2)="CS" THEN FLAG=1 :: CALL CLEAR ELSE FLAG=0
630 OPEN #1:FILE$,OUTPUT,DISPLAY ,FIXED LENGTH
640 FOR K=1 TO HILINE
650 PRINT #1:TEXT$(K)
660 NEXT K
670 CLOSE #1
680 IF FLAG THEN 170 ELSE 200
690 !NEU
700 FOR I=1 TO HILINE
710 TEXT$(I)=""
720 NEXT I
730 HILINE=0
740 GOTO 200
750 !FILENAME
760 ACCEPT AT(13,1)SIZE(-28)VALIDATE(DIGIT,UALPHA,"#."):FILE$
770 GOTO 220
780 !FEHLER MELDUNG
790 CALL ERR(CODE,TYPE)
800 IF CODE<>130 THEN RETURN
810 DISPLAY AT(23,1):"I/O ERROR ":"WEITER MIT <ENTER>"
820 CALL KEY(0,K,S):: IF S<=0 THEN 820
830 ON ERROR 850
840 CLOSE #TYPE
850 ON ERROR 780
860 DISPLAY AT(23,1): : :
870 IF FLAG THEN RETURN 170 ELSE RETURN 200
880 !AUFWIEDERSEHEN
890 CALL CLEAR
900 END
910 !ZEICHENFORMAT
920 DISPLAY AT(16,1):"1=80 / 2=64 ZEICHEN"
930 ACCEPT AT(16,21)SIZE(1)VALIDATE("12"):VALUE
940 IF VALUE=1 THEN LENGTH=80 :: LENGTH2=20 :: GOTO 960
950 LENGTH=64 :: LENGTH2=4
960 DISPLAY AT(16,21)
970 GOTO 210
```

Bild 3. Listing zu »Tagget-Code-Editor«. Damit können Tagget-Code-Files für TMS9900-Maschinenprogramme leichter eingegeben werden.

rät. Der Filename muß vorher unter (F)ilename definiert werden. (N)eu löscht den Speicherinhalt, und mit (A)ufwiedersehen verläßt man das Programm.

Wenn als Datenträger ein Kassettenrecorder herhalten muß, ist der Aufwand viel größer, weil hier das Datenformat FIXED 64 sein muß. Deshalb läßt sich der Editor mit (Z)eichenformat auf 64 Zeichen pro Zeile umschalten. Dazu muß der Tagget-Code umsortiert werden (siehe Bild 4). Grundsätzliches dazu: Der Tagget-Object-Code besteht aus einzelnen Tags. Jedes Programm beginnt mit dem Tag 0 zur Identifikation, gefolgt von einem maximal 8stelligen Wort. Dann folgt ein Adreßtag. An jedem Zeilenende muß ein Tag F stehen, um dem Loader das Ende einer Zeile zu melden. Die Prüfsummen mit Tag 8 oder Tag 7 sind nicht erforderlich. Wichtig ist aber noch der Tag 5, welcher die Eintrittsadresse festlegt, gefolgt von einem 6stelligen Namen. Das Ende eines Programmes erkennt der Loader und auch der Editor an einem Doppelpunkt, der am Anfang der letzten Zeile stehen muß, gefolgt von Leerzeichen oder einem Merktext. (M. Franz)

TAG TABELLE

TAG	Anwendung	Feld 1	Feld 2
0	Programm-Indentifizierung	Programm länge	Programm-Name
1	* Eintrittspunkt Definition	Absolute Adresse	
2	* Eintrittspunkt Definition	Relative Adresse	
3	* Externe Referenz	Relative Adresse	Symbol
4	* Externe Referenz	Absolute Adresse	Symbol
5	Externe Definition	Relative Adresse	Symbol
6	Externe Definition	Absolute Adresse	Symbol
7	Prüfsummen Indikator	Prüfsumme	
8	Prüfsummen ignorieren	irgendein Wert	
9	Lade Adresse	Absoluter Wert	
A	Lade Adresse	Relative Adresse	
B	Daten	Absoluter Wert	
C	Daten	Relative Adresse	
F	Ende der Zeile		

Die TAGs mit dem Sternchen sind für den Ext.-BASIC-Loader nicht geeignet.

Bild 4. Tag-Tabelle: Mit Anwendung, sowie Feld 1 und Feld 2.

Umlaut mit Funktionstaste

Im Juli-Heft schilderte ein Leser seine Schwierigkeiten mit dem kleinen »ä« beim Genie II. Diese lassen sich mit einem kurzen Maschinenprogramm leicht beseitigen. Mit dem Programm (26 Bytes) wird die F1-Taste zu »Ä« und »Shift F1« zu »ä«.

Video-Genie II

```
1 REM BASIC-LADER UMWANDLUNG F1-TASTE
2 A=65488       'A=ANFANGSADRESSE DES MASCHINENPROGRAMMS
                 BELIEBIG VERSCHIEBBAR, BEREICH SCHÜTZEN !
3 FOR P=A TO A+25
4 READ W:IF P>32767 THEN POKE P-65536,W ELSE POKE P,W
5 NEXT P:P=INT(A/256)
6 POKE16527,P:POKE16526,A-256*P:P=USR(Ø):GOTO 9 'LEVEL2-BASIC
7 IF A>32767 A=A-65536
8 DEFUSR1=A:P=USR1(Ø)                          'DISK-BASIC
9 DELETE 1-1Ø
1Ø DATA 42,22,64,34,222,255,33,221,255,34,22,64,2Ø1,2Ø5,Ø,Ø,254,
95,4Ø,3,254,127,192,214,4,2Ø1
```

Basic-Lader

```
FFDØ: 2A 16 4Ø 22 DE FF 21 DD FF 22 16 4Ø C9 CD ØØ ØØ
FFEØ: FE 5F 28 Ø3 FE 7F CØ D6 Ø4 C9 ØØ ØØ ØØ ØØ ØØ ØØ
```

Hex-Listing

```
ØØ1ØØ          ;BELEGUNG DER F1-TASTE GENIE II MIT A
ØØ11Ø          ORG     ØFFDØH          ;ODER WO MAN WILL
ØØ12Ø UMLAU    LD      HL,(4Ø16H)      ;LEITE
ØØ13Ø          LD      (DRIVER+1),HL   ;TASTATURABFRAGE
ØØ14Ø          LD      HL,DRIVER       ;       AUF NEUES
ØØ15Ø          LD      (4Ø16H),HL      ;          UNTERPROGRAMM
ØØ16Ø          RET
ØØ17Ø DRIVER   CALL    $-$             ;TESTE TASTATUR
ØØ18Ø          CP      5FH             ;IST ES F1-TASTE ?
ØØ19Ø          JR      Z,NEU           ;WENN JA, SPRINGE
ØØ2ØØ          CP      7FH             ;IST ES SHIFT-F1 ?
ØØ21Ø          RET     NZ              ;WENN NICHT, ZURÜCK
ØØ22Ø NEU      SUB     4               ;CODIERE UM AUF Ä
ØØ23Ø          RET
ØØ24Ø          END     UMLAU
```

Assembler-Listing

Das Programm ist im RAM-Bereich beliebig verschiebbar (Ladebereich schützen!). Falls ein Editor-Assembler vorhanden ist, kann das Assembler-Listing verwendet werden und über ORG der Ladebereich angegeben werden (zum Beispiel: 7FD0 für die 16-KByte-Version; 0FFD0 für die 48-KByte-Version. Ansonsten benutzt man am besten den Basic-Lader, wobei im Disk-Basic die Zeile 6 wegfallen muß. Falls Schwierigkeiten bei Zusatz-ROMs auftreten, sollte man erst nach Aufruf des ROMs dieses Programm benützen.

(Klaus Lingelmann)

TI 99/4A-Assembler in Extended Basic

Maschinensprache ist für schnelle Routinen beim TI 99/4A unentbehrlich. Das Editor-/Assembler-Paket oder das Minimem-Modul sind aber teuer. Für unser Programm sind nur nötig: ein Extended Basic-Modul und eine RAM-Erweiterung.

```
100 REM XBASIC-Assembler für RAM-Expansion
110 REM (c) Mag.Karl Hagenbuchner, Haidfeldstr. 2, A-4050 Traun; 1983-07-08
120 OPTION BASE 1
130 DIM OP$(64),OC$(64),J(64),LAB$(25),LAD$(25),REF$(15),RED$(15),VR$(15),VD$(15
)
135 ON ERROR 1000
140 DATA LI,020,8,AI,022,8,ANDI,024,8,ORI,026,8,CI,028,8,STWP,02A,9,STST,02C,9
150 DATA LWPI,02E0,11,LIMI,0300,11,RTWP,0380,10
160 DATA BLWP,04,6,B,044,6,X,048,6,CLR,04C,6,NEG,05,6,INV,054,6,INC,058,6,INCT,0
5C,6,DEC,06,6,DECT,064,6
170 DATA BL,068,6,SWPB,06C,6,SETO,07,6,ABS,074,6,SRA,08,5,SRL,09,5,SLA,0A,5,SRC,
0B,5
180 DATA JMP,10,2,JLT,11,2,JLE,12,2,JEQ,13,2,JHE,14,2,JGT,15,2,JNE,16,2,JNC,17,2
,JOC,18,2,JNO,19,2
190 DATA JL,1A,2,JH,1B,2,JOP,1C,2,SBO,1D,3,SBZ,1E,3,TB,1F,3,COC,20,4,CZC,24,4,XO
R,28,4,XOP,2C,4
200 DATA LDCR,30,4,STCR,34,4,MPY,38,4,DIV,3C,4,SZC,4,1,SZCB,5,1,S,6,1,SB,7,1,C,8
,1,CB,9,1
210 DATA A,A,1,AB,B,1,MOV,C,1,MOVB,D,1,SOC,E,1,SOCB,F,1
220 CALL CLEAR :: CALL SCREEN(11):: VV=0
230 DISPLAY AT(10,8):"XBasic-Assembler" :: DISPLAY AT(12,4):"Adressraum:)24F4 bi
s )3FFF" :: DISPLAY AT(24,9):"Bitte Warten !"
240 FOR I=1 TO 64 :: READ OP$(I),OC$(I),J(I):: NEXT I
250 CALL INIT :: CALL CLEAR :: L,F,V=1
255 INPUT "DRUCKER SYNCHRON (J/N):":DRUCK$ :: IF SEG$(DRUCK$,1,1)="J" THEN OPEN
```

Bild 1. Listing des Assemblers in Extended Basic

Das vorliegende Programm stellt einen recht komfortablen Line-by-Line-Assembler in Extended Basic dar, der dem des Minimem-Moduls ähnelt und die Erstellung von Maschinenprogrammen im RAM-Bereich ab >24F4 bis 3FFF erlaubt. Außerdem wurden noch Routinen eingebaut, die das Minimem-Modul nicht bietet, wohl aber der Editor/Assembler. Einige Hilfsroutinen gehen sogar über den Editor/Assembler hinaus. Nicht enthalten sind hingegen Möglichkeiten zum Editieren und Abspeichern des erzeugten Maschinencodes.

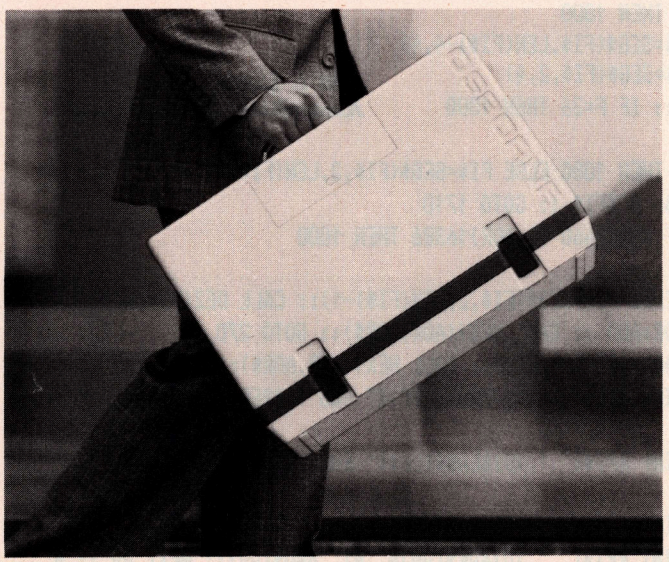
Die in diesem Artikel zu findenden Erläuterungen sind allerdings schon vom Umfang her nicht ausrei-

```
#1:"PIO" :: PRINT #1:CHR$(27);"B" :: DFL=1
260 CALL PEEK(8194,A,B):: CALL HEX(A,B,ADR$):: CALL DEZ(ADR$,ADR)
270 CALL PEEK(ADR,A,B):: CALL HEX(A,B,HEX$)
280 LINPUT ADR$&" "&HEX$&" ":IN$ :: T=0 :: IF IN$="" THEN ADR=ADR+2 :: CALL HEX1
(ADR,ADR$):: GOTO 270
290 IF SEG$(IN$,1,1)=" " THEN IN$=SEG$(IN$,2,LEN(IN$)-1):: GOTO 400
300 LAB$(L)=SEG$(IN$,1,POS(IN$," ",1)-1):: LAD$(L)=ADR$ :: L=L+1 :: IF L=21 THEN
 GOTO 1000
310 IN$=SEG$(IN$,POS(IN$," ",1)+1,LEN(IN$)-POS(IN$," ",1))
320 FOR VV=1 TO V :: IF VR$(VV)=LAB$(L-1)THEN CALL DEZ(VD$(VV),ADD):: GOSUB 350
330 NEXT VV :: GOTO 400
350 CALL DEZ(LAD$(L-1),Z):: CALL PEEK(ADD,A):: IF A=0 THEN CALL POKE(Z,P1,P2)::
CALL LOAD(ADD,P1,P2):: RETURN
360 ADD=ADD+2 :: DP=(Z-ADD)/2 :: IF DP<0 THEN DP=256+DP
370 CALL LOAD(ADD-1,DP):: RETURN
400 IF IN$="END" THEN 1100 ELSE IF IN$="NOP" THEN 1200 ELSE IF IN$="RT" THEN 130
0 ELSE IF IN$="RTWP" THEN 3900
405 IF IN$="REFLIST" THEN 2000 ELSE IF IN$="LABLIST" THEN 2010 ELSE IF IN$="DIS"
 THEN 2050 ELSE IF IN$="VORREF" THEN 2100
410 OPC$=SEG$(IN$,1,POS(IN$," ",1)-1):: FI$=SEG$(IN$,POS(IN$," ",1)+1,LEN(IN$)-P
OS(IN$," ",1))
420 IF OPC$="DATA" THEN 1400 ELSE IF OPC$="TEXT" THEN 1500 ELSE IF OPC$="EQU" TH
EN 1600
430 IF OPC$="AORG" THEN 1700 ELSE IF OPC$="BSS" THEN 1800
440 FOR I=1 TO 64 :: IF OPC$=OP$(I)THEN 460
450 NEXT I :: DISPLAY AT(24,2):"SYNTAXERROR" :: GOTO 270
460 ON J(I)GOSUB 3000,3100,3200,3300,3400,3500,3600,3700,3800,3900,4000
```

Bild 1. Listing des Assemblers (Fortsetzung)

Osborne. Der erste Portable.

Bis Mitte 1981 gab es keine tragbaren Computer. Bis dahin waren Computer entweder zu groß und zu schwer. Oder Computer waren zu klein und nicht ausreichend nutzbar.



Und die Computer-Industrie machte keine Anstalten, die Lücke zu füllen. Die Innovation kam schließlich von einem Neuling und verblüffte die gesamte Branche. Osborne brachte Mitte 1981 den ersten tragbaren Compact-Computer auf den Markt.
Ein revolutionäres System mit hervorragenden Leistungsdaten. Mit zwei Diskettenlaufwerken, eingebautem Monitor, Textverarbeitungs- und Rechensystemen sowie den Programmiersprachen CBASIC und MBASIC. Ein vollwertiger Computer und mit nur 12 kg trotzdem ein Leichtgewicht. Transportablel wie ein Aktenkoffer und sogar netzunabhängig einzusetzen. Osborne hat mit Erfolg einen neuen Markt geschaffen. Mit dem ersten Computer als Portable.

OSBORNE
Management by Computer.

chend, um mehr als erste Experimente zu ermöglichen. Für weiterreichende Anwendungen muß man sich das Begleitbuch des Editor/Assembler-Pakets besorgen, das es auch gesondert zu kaufen gibt. Noch ein Rat vorweg: Sichern Sie nach dem Eintippen des Assemblers aus Bild 1 dieses Programm unbedingt auf Band oder Diskette, bevor Sie es zum ersten Mal laufen lassen. Maschinencode hat die fatale Eigenschaft, immer im ungeeignetsten Augenblick das Rechnersystem auf ein totes Gleis zu schieben, von dem es nur noch mit Reset zurückgeholt werden kann. Aber dann ist jedes Programm unwiederbringlich verloren.

Vor dem ersten Start: Programm abspeichern

Das Programm des Assemblers kann ohne jede Vorkenntnis über die Zusammenhänge in den Computer eingegeben werden. Nach dem Programmstart meldet sich das Programm dann mit der niedrigsten verfügbaren Programmadresse >24F4. Die Eingabe der Directiven erfolgt in der Form: »(Label) <Opcode> <Operandfeld>«, wobei runde Klammern optionale und eckige Klammern notwendige Angaben kennzeichnen. Wird kein Label gesetzt, ist ein führendes Leerzeichen einzugeben, ebenso werden Opcode und Operanden durch ein Leerzeichen getrennt. Kommentare sind nicht zulässig. Labels dürfen die Länge von 25 Zeichen nicht überschreiten. Auch Definitionen sind auf diese Anzahl limitiert. Die Anzahl möglicher Vorwärtsreferenzen ist auf 15 begrenzt; es gibt keine apriori-Definitionen wie beim Editor/Assembler. Definiert wird im Stil des Minimem mit »Label EQU Adresse«. Alle Programme müssen mit Absolutadressen eingegeben werden — sie sind also nicht relocatibel. Der Befehl RORG fehlt. Pro DATA-Befehl ist nur eine Zahl zulässig. Da auch eine nachträgliche Editiermöglichkeit fehlt (außer einem Neuaufrollen

```
470 CALL DEZ(ADR$,ADR):: DDR$=ADR$ :: CALL DEZ(HEX$,HEX):: CALL POKE(HEX,P1,P2):
: CALL LOAD(ADR,P1,P2):: ADR=ADR+2 :: CALL HEX1(ADR,ADR$)
480 DISPLAY AT(23,6)SIZE(4):HEX$ :: IF DFL=1 THEN PRINT #1:DDR$&" "&HEX$&" "&IN$
485 IF T=0 THEN 270
490 CALL DEZ(HE$,HE):: CALL POKE(HE,P1,P2):: CALL LOAD(ADR,P1,P2)
495 IF DFL=1 THEN PRINT #1:ADR$&" "&HE$
500 PRINT ADR$&" "&HE$ :: HE$="" :: ADR=ADR+2 :: CALL HEX1(ADR,ADR$):: IF T=1 TH
EN 270
510 CALL DEZ(H$,H):: CALL POKE(H,P1,P2):: CALL LOAD(ADR,P1,P2):: PRINT ADR$&" "&
H$ :: DDD$=ADR$ :: ADR=ADR+2 :: CALL HEX1(ADR,ADR$)
520 IF DFL=1 THEN PRINT #1:DDD$&" "&H$
530 GOTO 270
1000 IF L<25 AND V<15 AND F<15 THEN PRINT "          EINGABEFEHLER !" :: GOTO 27
0 ELSE 1010
1010 IF L>25 THEN PRINT "LABELBEREICH AUSGESCHOEPFT!" :: GOTO 270
1020 PRINT "ZU VIELE REFERENZEN !"
1030 GOTO 270
1100 CALL POKE(ADR,P1,P2):: CALL LOAD(8194,P1,P2):: CALL PEEK(8196,A,B):: CALL H
EX(A,B,HEX$)
1110 CALL DEZ(HEX$,DEZ):: DEZ=DEZ-8 :: CALL HEX1(DEZ,ADR$):: CALL POKE(DEZ,P1,P2
):: CALL LOAD(8196,P1,P2)
1120 INPUT "Programmname:":NAME$ :: IF LEN(NAME$)>6 THEN NAME$=SEG$(NAME$,1,6)
1130 IF LEN(NAME$)<6 THEN NAME$=NAME$&RPT$(" ",6-LEN(NAME$))
1140 FOR I=1 TO 6 :: CALL LOAD(DEZ-1+I,ASC(SEG$(NAME$,I,1))):: NEXT I
1150 INPUT "Programmstartadresse:":START$
1160 IF SEG$(START$,1,1)=")" THEN IF LEN(START$)=5 THEN CALL DEZ(SEG$(START$,2,4
),ADR):: CALL POKE(ADR,P1,P2):: CALL LOAD(DEZ+6,P1,P2):: GOTO 1192
1170 FOR LL=1 TO L :: IF LAB$(LL)=START$ THEN 1190
1180 NEXT LL :: PRINT "LABEL NICHT GEFUNDEN!" :: GOTO 1150
1190 CALL DEZ(LAD$(LL),DE):: CALL POKE(DE,P1,P2):: CALL LOAD(DEZ+6,P1,P2)
1192 INPUT "NOCH EIN NAME (J/N)? ":J$ :: IF J$<>"J" THEN END ELSE 1100
1200 HEX$="1000" :: T=0 :: GOTO 470
1300 HEX$="0458" :: T=0 :: GOTO 470
1400 IF SEG$(FI$,1,1)<>")" THEN CALL HEX1(VAL(FI$),HEX$):: GOTO 470
1410 FI$=SEG$(FI$,2,LEN(FI$)-1)
1420 IF LEN(FI$)>4 THEN FI$=SEG$(FI$,LEN(FI$)-4,4)
1430 IF LEN(FI$)<4 THEN FI$="0"&FI$ :: GOTO 1430
1440 HEX$=FI$ :: GOTO 470
1500 FI$=SEG$(FI$,2,LEN(FI$)-2):: IF POS(FI$,"'",1)THEN FI$=SEG$(FI$,1,POS(FI$,"
'",1)-1)
1505 IF INT(LEN(FI$)/2)*2<>LEN(FI$)THEN FI$=FI$&" "
1510 FOR S=1 TO LEN(FI$)STEP 2 :: A$=ASC(SEG$(FI$,S,1)):: CALL HEX1(A$,A$):: A
$=SEG$(A$,3,2):: A$=ASC(SEG$(FI$,S+1,1)):: CALL HEX1(A$,AB$)
1520 CALL LOAD(ADR,ASC(SEG$(FI$,S,1)),ASC(SEG$(FI$,S+1,1))):: PRINT ADR$&" "&A$
&SEG$(AB$,3,2)
1530 ADR=ADR+2 :: CALL HEX1(ADR,ADR$):: NEXT S :: GOTO 270
1600 REF$(F)=LAB$(L-1):: L=L-1 :: LAB$(L)=""
1610 IF SEG$(FI$,1,1)<>")" THEN 1000
1620 IF LEN(FI$)>5 THEN FI$=SEG$(FI$,LEN(FI$)-4,4)
1625 IF LEN(FI$)=5 THEN FI$=SEG$(FI$,2,4)
1630 RED$(F)=FI$ :: F=F+1 :: IF F=26 THEN 1000
1640 GOTO 270
1700 IF SEG$(FI$,1,1)<>")" THEN 1000 ELSE FI$=SEG$(FI$,2,LEN(FI$)-1)
1710 IF LEN(FI$)<4 THEN FI$="0"&FI$ :: GOTO 1710
1720 CALL DEZ(FI$,ADR):: IF ADR<9460 OR ADR>16386 THEN 1000
1730 ADR$=FI$ :: GOTO 270
1800 IF SEG$(FI$,1,1)=")" THEN FI$=SEG$(FI$,2,LEN(FI$)-1):: CALL DEZ(FI$,ADD)::
CALL DEZ(ADR$,ADR):: ADR=ADR+ADD :: CALL HEX1(ADR,ADR$):: GOTO 270
1810 CALL DEZ(ADR$,ADR):: ADR=ADR+VAL(FI$):: CALL HEX1(ADR,ADR$):: GOTO 270
2000 FOR FF=1 TO F-1 :: PRINT RPT$(" ",10)&REF$(FF)&" EQU )"&RED$(FF):: NEXT FF
:: GOTO 270
2010 FOR LL=1 TO L-1 :: PRINT RPT$(" ",10)&LAB$(LL)&" EQU )"&LAD$(LL):: NEXT LL
:: GOTO 270
2050 PRINT "DISASSEMBLER EINLEGEN !" :: RUN "CS1"
2100 FOR VV=1 TO V-1 :: PRINT RPT$(" ",10)&VR$(VV)&" AT "&VD$(VV):: NEXT VV :: G
OTO 270
3000 CALL BIN(OC$(I),BIN$):: BIN$=SEG$(BIN$,1,4):: G$=FI$ :: FI$=SEG$(FI$,POS(FI
```

Bild 1. Listing des Assemblers (Fortsetzung)

mit AORG), besteht die Möglichkeit, jede Eingabe vom Drucker gleich dokumentieren zu lassen. Die Ausdruckroutine ist in der vorliegenden Form auf den Drucker GP700A eingestellt (Anschluß über parallele Schnittstelle).

Der Befehlssatz ist fast so umfangreich wie beim Minimem

Die vom Assembler unterstützten Direktiven und den Befehlssatz zeigen Tabellen in der nächsten Ausgabe.

Um den Programmzähler nicht mehrmals umstellen zu müssen, enthält der Extended Basic-Assembler eine bequeme Benennungsroutine, die mit der END-Instruktion aktiviert wird. Programme werden vom Extended Basic-Loader nur dann gefunden, wenn die Adresse >2002 die erste freie Adresse im RAM Bereich >2000->3FFF enthält und auf >2004 der Beginn der Namenstabelle angezeigt wird. Für jeden Programmnamen sind bis zu sechs Zeichen zulässig, dem die Programmbeginn-Adresse zu folgen hat. Sich darum kümmern zu müssen, nimmt einem der Extended Basic-Assembler auf folgende Art ab: Nach Eingabe von »END« wird der Name des Programms erfragt (der intern eventuell auf sechs Zeichen erweitert oder gekürzt wird) und an die richtige Stelle geladen. Gleichzeitig werden in die Adressen >2002 und >2004 die zugehörigen Werte geschrieben und nach der Programmstart-Adresse gefragt. Hierbei ist der Label und keine Hexadresse anzugeben. Die Adresse wird automatisch der Labeltabelle entnommen. Nach dem ersten Programmnamen können weitere eingegeben werden, solange die jeweilige Frage nach weiteren Programmnamen bejaht wird. Nach dem endgültigen Abschluß können die Maschinenprogramme als Teil von Extended Basic-Programmen oder im Direktmodus mittels »CALL LINK("Programmname",Parameterliste)« angesprochen werden.

Außer den bisher erwähnten Unterschieden (nur eine

```
$,",",1)+1,LEN(FI$)-POS(FI$,",",1))
3010 GOSUB 3600 :: FI$=SEG$(G$,1,POS(G$,",",1)-1):: GOSUB 3600 :: CALL HEX2(BIN$
,HEX$):: RETURN
3100 HEX$=0C$(I):: IF SEG$(FI$,1,1)()")" THEN GOSUB 10000 :: GOSUB 3720 ELSE HE$
=SEG$(FI$,2,LEN(FI$)-1):: GOSUB 3720
3110 CALL DEZ(HE$,DP):: DP=(DP-ADR-2)/2 :: IF DP<0 THEN DP=256+DP
3120 CALL HEX1(DP,HE$):: HEX$=HEX$&SEG$(HE$,3,2):: RETURN
3200 CALL HEX1(VAL(FI$),HE$):: HEX$=0C$(I)&SEG$(HE$,3,2):: RETURN
3300 CALL BIN(0C$(I),BIN$):: BIN$=SEG$(BIN$,1,6):: G$=SEG$(FI$,POS(FI$,",",1)+1,
LEN(FI$)-POS(FI$,",",1))
3310 CALL HEX1(VAL(G$),HE$):: CALL BIN(HE$,BI$):: BIN$=BIN$&SEG$(BI$,LEN(BI$)-3,
4)
3320 FI$=SEG$(FI$,1,POS(FI$,",",1)-1):: GOTO 3510
3400 F$=SEG$(FI$,1,POS(FI$,",",1)-1):: FI$=SEG$(FI$,POS(FI$,",",1)+1,LEN(FI$)-PO
S(FI$,",",1))
3410 CALL HEX1(VAL(FI$),A$):: HEX$=0C$(I)&SEG$(A$,4,1):: CALL HEX1(VAL(F$),A$)::
 HEX$=HEX$&SEG$(A$,4,1):: RETURN
3500 CALL BIN(0C$(I),BIN$):: BIN$=SEG$(BIN$,1,10):: GOSUB 3510 :: RETURN
3510 IF SEG$(FI$,1,1)="*" THEN 3540 ELSE IF SEG$(FI$,1,1)="@" THEN 3560
3520 IF ASC(SEG$(FI$,1,1))<65 THEN BIN$=BIN$&"00" ELSE 1000
3530 CALL HEX1(VAL(FI$),A$):: CALL HEX2(BIN$,B$):: HEX$=SEG$(B$,1,3)&SEG$(A$,4,1
):: RETURN
3540 IF SEG$(FI$,LEN(FI$),1)="+" THEN BIN$=BIN$&"11" :: GOTO 3550 ELSE BIN$=BIN$
&"01" :: FI$=SEG$(FI$,2,LEN(FI$)-1):: GOTO 3530
3550 FI$=SEG$(FI$,2,LEN(FI$)-2):: GOTO 3530
3560 BIN$=BIN$&"10" :: T=1 :: FI$=SEG$(FI$,2,LEN(FI$)-1):: IF POS(FI$,"(",1)THEN
 FF$=SEG$(FI$,1,POS(FI$,"(",1)-1):: GOTO 3580
3565 IF SEG$(FI$,1,1)=")" THEN HE$=SEG$(FI$,2,LEN(FI$)-1):: FI$="0" :: GOTO 3530
3570 IF ASC(SEG$(FI$,1,1))<65 THEN CALL HEX1(VAL(FI$),HE$):: FI$="0" :: GOTO 353
0 ELSE GOSUB 10000 :: GOSUB 3720 :: FI$="0" :: GOTO 3530
3580 FI$=SEG$(FI$,POS(FI$,"(",1)+1,LEN(FI$)-POS(FI$,"(",1)-1):: IF SEG$(FF$,1,1)
=")" THEN HE$=SEG$(FF$,2,LEN(FF$)-1):: GOTO 3530
3590 IF ASC(SEG$(FF$,1,1))<65 THEN CALL HEX1(VAL(FF$),HE$):: GOTO 3530 ELSE FG$=
FI$ :: FI$=FF$ :: GOSUB 10000 :: GOSUB 3720 :: FI$=FG$
3595 GOTO 3530
3600 IF SEG$(FI$,1,1)="*" THEN 3630 ELSE IF SEG$(FI$,1,1)="@" THEN 3650
3610 IF ASC(SEG$(FI$,1,1))<65 THEN BIN$=BIN$&"00" ELSE 1000
3620 CALL HEX1(VAL(FI$),B$):: CALL BIN(B$,BI$):: BI$=SEG$(BI$,LEN(BI$)-3,4):: BI
N$=BIN$&BI$
3625 IF T=2 THEN X$=HE$ :: HE$=H$ :: H$=X$ :: RETURN ELSE RETURN
3630 IF SEG$(FI$,LEN(FI$),1)="+" THEN BIN$=BIN$&"11" :: GOTO 3640 ELSE BIN$=BIN$
&"01" :: FI$=SEG$(FI$,2,LEN(FI$)-1):: GOTO 3620
3640 FI$=SEG$(FI$,2,LEN(FI$)-2):: GOTO 3620
3650 BIN$=BIN$&"10" :: T=T+1 :: FI$=SEG$(FI$,2,LEN(FI$)-1):: IF POS(FI$,"(",1)TH
EN FF$=SEG$(FI$,1,POS(FI$,"(",1)-1):: GOTO 3680
3660 IF SEG$(FI$,1,1)=")" AND T=1 THEN HE$=SEG$(FI$,2,LEN(FI$)-1):: FI$="0" :: G
OTO 3620 ELSE IF SEG$(FI$,1,1)=")" AND T=2 THEN H$=SEG$(FI$,2,LEN(FI$)-1):: FI$=
"0" :: GOTO 3620
3670 IF ASC(SEG$(FI$,1,1))<65 THEN CALL HEX1(VAL(FI$),K$):: FI$="0" :: IF T=1 TH
EN HE$=K$ :: GOTO 3520 ELSE IF T=2 THEN H$=K$ :: GOTO 3520
3675 GOSUB 10000 :: GOSUB 3720 :: FI$="0" :: GOTO 3620
3680 FI$=SEG$(FI$,LEN(FF$)+2,LEN(FI$)-LEN(FF$)-2):: IF SEG$(FF$,1,1)=")" THEN IF
 T=1 THEN HE$=SEG$(FF$,2,LEN(FF$)-1):: GOTO 3520 ELSE H$=SEG$(FF$,2,LEN(FF$)-1):
: GOTO 3520
3690 IF ASC(SEG$(FF$,1,1))<65 THEN CALL HEX1(VAL(FF$),K$):: IF T=1 THEN HE$=K$ :
: GOTO 3520 ELSE IF T=2 THEN H$=K$ :: GOTO 3520
3695 FG$=FI$ :: FI$=FF$ :: GOSUB 10000 :: GOSUB 3720 :: FI$=FG$ :: GOTO 3620
3700 F$=SEG$(FI$,1,POS(FI$,",",1)-1):: FI$=SEG$(FI$,POS(FI$,",",1)+1,LEN(FI$)-PO
S(FI$,",",1)):: CALL HEX1(VAL(F$),A$)
3710 HEX$=0C$(I)&SEG$(A$,4,1):: T=1 :: IF SEG$(FI$,1,1)=")" THEN HE$=SEG$(FI$,2,
LEN(FI$)-1):: GOTO 3720 ELSE IF ASC(SEG$(FI$,1,1))<65 THEN 3715 ELSE 3740
3715 IF SEG$(FI$,1,1)="-" THEN CALL BIN2(VAL(FI$),BI$):: CALL HEX2(BI$,HE$)ELSE
```

Bild 1. Listing des Assemblers (Fortsetzung)

```
CALL HEX1(VAL(FI$),HE$)
3720 IF LEN(HE$)>4 THEN HE$=SEG$(HE$,LEN(HE$)-4,4)ELSE IF LEN(HE$)<4 THEN HE$=RP
T$("0",4-LEN(HE$))&HE$
3730 RETURN
3740 GOSUB 10000 :: GOTO 3720
3800 CALL HEX1(VAL(FI$),A$):: HEX$=OC$(I)&SEG$(A$,4,1):: RETURN
3900 HEX$="0380" :: GOTO 470
4000 HEX$=OC$(I):: T=1 :: IF SEG$(FI$,1,1)<>")" AND OP$(I)="LWPI" THEN 10000
4010 IF SEG$(FI$,1,1)=")" THEN HE$=SEG$(FI$,2,LEN(FI$)-1):: GOTO 3720
4020 HE$=FI$ :: GOTO 3720
10000 FOR LL=1 TO L :: IF LAB$(LL)=FI$ THEN IF T=1 THEN HE$=LAD$(LL):: RETURN EL
SE H$=LAD$(LL):: RETURN
10010 NEXT LL :: FOR FF=1 TO F :: IF REF$(FF)=FI$ THEN IF T=1 THEN HE$=RED$(FF):
: RETURN ELSE H$=RED$(FF):: RETURN
10020 NEXT FF :: GOSUB 12000 :: IF HE$="" THEN 1000 ELSE RETURN
12000 VR$(V)=FI$ :: IF SEG$(OP$(I),1,1)="J" THEN CALL HEX1(ADR,ADD$)ELSE CALL HE
X1(ADR+2*T,ADD$)
12010 VD$(V)=ADD$ :: V=V+1 :: HE$="0" :: IF V>15 THEN 1000 ELSE RETURN
20000 SUB HEX1(ZAHL,HEX$)
20010 Z1=INT(ZAHL/(16*16*16)):: Z2=ZAHL-Z1*16*16*16 :: Z3=INT(Z2/(16*16)):: Z4=Z
2-Z3*16*16
20020 Z5=INT(Z4/16):: Z6=Z4-Z5*16
20030 IF Z1>9 THEN HEX$=CHR$(55+Z1)ELSE HEX$=STR$(Z1)
20040 IF Z3>9 THEN HEX$=HEX$&CHR$(55+Z3)ELSE HEX$=HEX$&STR$(Z3)
20050 IF Z5>9 THEN HEX$=HEX$&CHR$(55+Z5)ELSE HEX$=HEX$&STR$(Z5)
20060 IF Z6>9 THEN HEX$=HEX$&CHR$(55+Z6)ELSE HEX$=HEX$&STR$(Z6)
20070 SUBEND
20080 SUB HEX(A,B,HEX$)
20090 CALL HEX1(A,HEX$):: H$=SEG$(HEX$,3,2):: CALL HEX1(B,HEX$):: HEX$=SEG$(HEX$
```

Bild 1. Listing des Assemblers (Fortsetzung)

Zahl pro DATA, außer Indizierung keine Berechnungen im Operandenfeld, keine relocatiblen Programme, keine vorgegebenen Definitionen) kennt der Extended Basic-Assembler auch nicht die Verwendung des Symbols »$« als Synonym für den Wert des Programm-Counters. Eine Anweisung JMP $+4 wird daher gleich aus zwei Gründen nicht erkannt. Ansonsten werden alle Adressierungsarten unterstützt.

Der Hauptunterschied aber ist selbstverständlich die Verarbeitungsgeschwindigkeit. Der Extended Basic-Assembler muß ja nicht nur durch eine Analyse der Eingabe die an die betreffenden Speicherstellen zu pokenden Werte ermitteln, sondern ist auch noch mit der Erstellung eines ansprechenden Bildschirms zur Dokumentation der Befehle befaßt, da der jeweilige Inhalt der Adressen nach der Assemblierung angezeigt wird. Gemeinsam mit dem Minimem, aber im Unterschied zum Editor-Assem-

Osborne Executive. Der Business-Computer.

Osborne Executive heißt der Business-Computer für alle, die die Produktivität noch weiter steigern müssen. Dazu ist der Osborne Executive mit einem erweiterten, leistungsfähigen System an serienmäßiger Hardware ausgestattet.

- Z-80 A CPU, 4 MHz. Mikroprozessor, 128 KB Arbeitsspeicher slim-line.
- Zwei Floppy-Laufwerke à 200 KB (186.368 Zeichen). Das entspricht über 100 Schreibmaschinenseiten.
- Ablagefächer für Disketten.
- Formatkompatibel mit IBM PC, DEC VT 180, XEROX 821, CROMMENCO und UCSD-PASCAL-SYSTEM-Universalformat.
- Klar lesbarer, entspiegelter 7-Zoll-Bildschirm (bernsteinfarbene Darstellung). Sichtbares Feld von 24 Zeilen à 80 Zeichen. Nahezu alle Videoattribute, frei definierbarer Zeichensatz, programmierbares Blinken, Unterstreichung, halbe Intensität.
- Deutsche und internationale Schreibmaschinentastatur, mit einem separaten, frei programmierbaren Zehnerblock und zusätzlichen Cursorsteuertasten.
- Zwei serielle Schnittstellen über Z 80 SIO (synchron und asynchron), Anschlußmöglichkeit für Drucker, Akustikkoppler und andere Computer.
- Optional Terminal-Emulation und synchrone Großrechnerprozeduren (z. B. Siemens MSV 1 und IBM 3270).
- Aufrüstbar auf 256 KB RAM, 16 Bit, MS-DOS, CP/M-86.
- Anschluß an nahezu alle Stromnetze.
- Wetterfestes, spritzwassergeschütztes, tragbares Gehäuse.

OSBORNE

Management by Computer.

bler, besteht keine Möglichkeit, Maschinenprogramme direkt auf Diskette speichern zu können. Wenn die Programme nicht wieder herausgepeekt und zu einer Poke-Liste verarbeitet werden, sind sie nach dem Ausschalten der Peripheriebox verloren. Beim Ausschalten der Konsole bleiben sie in der RAM-Erweiterung aber erhalten.

Der Extended Basic-Assembler eröffnet aber auf alle Fälle eine kostengünstige Möglichkeit zur Erstellung von Maschinenprogrammen ohne zusätzliches Kommandomodul, wenn die RAM-Expansion und das Extended Basic-Modul als notwendige Voraussetzung vorhanden sind.

Aus Platzgründen bringen wir die Tabellen mit den Adressierungsarten, Direktiven und Befehlen zusammen mit einem Programm zur Abspeicherung von assemblierten Programmen in Ausgabe 23.

(Karl Hagenbuchner)

```
,3,2):: HEX$=H$&HEX$
20100 SUBEND
20110 SUB DEZ(HM$,DD)
20120 DD=0 :: IF LEN(HM$)<4 THEN HM$="0"&HM$ :: GOTO 20120
20130 FOR I=1 TO 4 :: IF ASC(SEG$(HM$,I,1))>57 THEN DD=DD+16^(4-I)*((ASC(SEG$(HM
$,I,1))-55))ELSE DD=DD+16^(4-I)*VAL(SEG$(HM$,I,1))
20140 NEXT I
20150 SUBEND
20160 SUB POKE(D,P1,P2):: CALL HEX1(D,DD$):: D1$=SEG$(DD$,1,2):: D2$=SEG$(DD$,3,
2):: D1$="00"&D1$ :: D2$="00"&D2$
20170 CALL DEZ(D1$,P1):: CALL DEZ(D2$,P2)
20180 SUBEND
20190 SUB BIN(HE$,BIN$)
20195 IF LEN(HE$)<4 THEN HE$=HE$&"0" :: GOTO 20195
20200 CALL DEZ(HE$,DD):: J=15 :: IF DD>32767 THEN BIN$="1" :: DD=DD-32768 ELSE B
IN$="0"
20210 IF DD>16383 THEN BIN$=BIN$&"1" :: DD=DD-16384 :: J=14
20220 FOR I=J TO 1 STEP -1 :: BIN$=BIN$&STR$(ABS(SGN(DD AND 2^(I-1)))):: NEXT I
20230 SUBEND
20240 SUB HEX2(BI$,B$)
20250 B=0 :: FOR I=1 TO LEN(BI$):: IF SEG$(BI$,I,1)="1" THEN B=B+2^(16-I)
20260 NEXT I :: CALL HEX1(B,B$)
20270 SUBEND
20280 SUB BIN2(ZA,BIN$)
20290 ZB=ZA :: BIN$="1" :: IF ZA<-16383 THEN ZA=ZA+16384
20300 IF ZB<-16383 THEN BIN$=BIN$&"0" ELSE IF ZA<0 THEN BIN$=BIN$&"1"
20310 FOR I=14 TO 1 STEP -1 :: BIN$=BIN$&STR$(ABS(SGN(ZA AND 2^(I-1)))):: NEXT I
 :: SUBEND
```

Bild 1. Listing des Assemblers (Schluß)

Osborne. Teure Software inklusive.

Osborne hat die Computer, die Routine schnell erledigen und schwierige Probleme sofort lösen. Darüber hinaus – und das ist ungewöhnlich – liefert Osborne mehrere Software-Pakete inklusive mit. Software-Pakete, die normalerweise als teure Extras gekauft werden müssen.

Software inklusive für den Osborne 1.

- Betriebssoftware:
  - CP/M 2.2
  - CBASIC (p-Code Compiler)
  - MBASIC (BASIC 80 interpreter)
- Deutsche Anwendersoftware:
  - WORDSTAR Textverarbeitungsprogramm
  - MAILMERGE Adressenzumischung
  - SUPERCALC Rechenprogramm
  - FIBU Finanzbuchhaltung (dt. Recht)

Software inklusive für den Osborne Executive.

- Betriebssoftware:
  - CP/M Plus
  - UCSD-PASCAL Betriebssystem
  - CBASIC (p-Code Compiler)
  - MBASIC (BASIC 80 interpreter)
- Anwendersoftware:
  - WORDSTAR Textverarbeitungsprogramm
  - MAILMERGE Adressenzumischung
  - SUPERCALC Rechenprogramm
  - PERSONAL PEARL Datenbanksystem mit Masken- und Druckgenerator. Der Anwender kann z. B. Lager- und Kundenverwaltung selbst erstellen. Personal Pearl kann mit Wordstar und Supercalc verbunden werden.
  - FIBU Finanzbuchhaltung (dt. Recht)
  - TOPTIP rechnende Textverarbeitung

OSBORNE
Management by Computer.

TI 99/4A-Videothek

Sobald man eine gewisse Anzahl Videocassetten besitzt, wird es etwas kompliziert, zum Beispiel einen bestimmten Film herauszusuchen. Das Programm »Videothek« soll allen, die den TI 99/4A mit Extended Basic besitzen, die Suche erleichtern.

Im Gegensatz zu vielen sogenannten »Luxusprogrammen« beschränkt sich dieses Programm auf das Notwendigste und stellt so eine praktische Hilfe für den Videofan dar. Die ganze Datei ist menügesteuert und erklärt sich beim Betrieb von selbst. Dennoch soll das Programm hier kurz erläutert werden.

Im Menü kann man wählen zwischen den Programmteilen »laden«, »speichern«, »drucken«, »schreiben«, »ergänzen«, »ändern« und »suchen«. Die Option »laden« wird gewöhnlich als Erstes gewählt, wenn bereits eine Datei auf Kassette existiert. Es können in einer Datei bis zu 50 Kassetten mit jeweils sechs Datenangaben (Titel, Bandnummern, zwei Schauspielernamen und zwei Suchbegriffe) gespeichert werden. Die Eingabe der Zahl 8 im Menü stoppt das Programm.

»Speichern« wird gewählt, wenn die eingetippten Daten auf einer Kassettendatei gespeichert werden sollen. Vor der Eröffnung einer Datei wird das Unterprogramm »Sort« aufgerufen, welches die Filmtitel alphabetisch ordnet. Anschließend erfolgt die Speicherung. »Drucken« veranlaßt den Computer, eine Hardcopy des Dateiinhalts auf dem Drucker auszugeben. Auch hier wird vor dem Druck die Sort-Routine aufgerufen. »Schreiben« ermöglicht es, Daten in die Datei einzugeben. Die Speicherung erfolgt ab Datensatz 1; eventuell bereits vorhandene Daten werden dabei gelöscht. Um Datensätze anhängen zu können, muß »ergänzen« gewählt werden. Bei dieser Wahl wird der letzte Datensatz gesucht und nach dessen Erreichen zum Teil »schreiben« verzweigt. Hierbei erscheinen auf dem Bildschirm jeweils die Eingabekriterien (Titel, Bandnummer, etc.). Angenommen werden Eingaben mit maximal 28 Zeichen.

»Ändern« bietet zwei Möglichkeiten: Das Ändern eines Bandinhalts, wenn zum Beispiel ein Film gelöscht wurde, oder das Berichtigen von Fehlern. Bei ersterem muß nur die betreffende Bandnummer eingegeben werden und der alte Inhalt des Bandes erscheint zu Änderungszwecken auf dem Bildschirm. Im zweiten Fall muß darüber hinaus noch eingegeben werden, wo ein Fehler vorliegt (zum Beispiel falscher Schauspielername). Hierauf erscheinen der Reihe nach alle Datensätze, wobei man jeweils zwischen »nächster Datensatz«, »ändern« und »Ende — zurück zum Menü« wählen kann.

»Suchen« schließlich ermöglicht es, zum Beispiel einen bestimmten Filmtitel oder einen Film mit einem bestimmten Schauspieler zu suchen. Wird ein Film nicht gefunden, oder entspricht nach der Anweisung »weitersuchen« kein weiterer Datensatz dem eingegebenen Begriff, erscheint »nicht gefunden« auf dem Bildschirm. Der Aufruf des Unterprogramms »Delay« erzeugt eine kurze Pause, bevor das Menü von Neuem erscheint.

```
100 REM VIDEODATEI
110 REM BY ALWIN ERTL
120 REM    MAIS 66
130 REM    8497 NEUKIRCHEN
140 DIM T$(50,6)
150 REM MENUE
160 BF=16 :: RF=13 :: GOSUB 14
    00
170 DISPLAY AT(1,9):"MENUE..."
     :: DISPLAY AT(3,1):"EINGA
    BE  FUER"
180 FOR I=1 TO 6 :: DISPLAY AT
    (3+2*I,4):I :: DISPLAY AT(
    3+2*I,10):"DATEI" :: NEXT
    I
190 DATA LADEN,SPEICHERN,DRUCK
    EN,SCHREIBEN,ERGAENZEN,AEN
    DERN
200 RESTORE 190 :: FOR I=1 TO
    6 :: READ A$ :: DISPLAY AT
    (3+2*I,16):A$ :: NEXT I ::
     DISPLAY AT(17,5):"7   IN
     DATEI SUCHEN"
210 DISPLAY AT(18,7):"IHRE WAH
    L:" :: ACCEPT AT(20,12)BEE
    P("12345678")SIZE(1):A
220 ON A GOTO 230,290,340,400,
    520,550,1030,1520
230 REM LADEN
240 BF=8 :: RF=10 :: GOSUB 140
    0
250 OPEN #1:"CS1",INTERNAL ,IN
    PUT ,FIXED 192
260 FOR I=1 TO 50 :: INPUT #1:
    T$(I,1),T$(I,2),T$(I,3),T$
    (I,4),T$(I,5),T$(I,6)
270 IF T$(I,1)="ZZZ" THEN CLOS
    E #1 :: GOTO 150
280 NEXT I :: CLOSE #1 :: GOTO
     150
290 BF=10 :: RF=8 :: GOSUB 140
    0 !SPEICHERN
300 CALL SORT(T$(,)):: OPEN #1
    :"CS1",INTERNAL ,OUTPUT ,F
    IXED 192
310 FOR I=1 TO 50 :: PRINT #1:
    T$(I,1),T$(I,2),T$(I,3),T$
    (I,4),T$(I,5),T$(I,6)
320 IF T$(I,1)="ZZZ" THEN CLOS
    E #1 :: GOTO 150
330 NEXT I :: CLOSE #1 :: GOTO
     150
340 BF=15 :: RF=11 :: GOSUB 14
    00 !DRUCKEN
350 CALL SORT(T$(,)):: OPEN #1
    :"TP.E",OUTPUT
360 FOR I=1 TO 50 :: FOR J=1 T
    O 6
370 IF T$(I,J)="ZZZ" THEN CLOS
    E #1 :: GOTO 150
380 PRINT #1:T$(I,J):: NEXT J
    :: PRINT #1:"" :: NEXT I
390 CLOSE #1 :: GOTO 150
400 Z=0 !SCHREIBEN
410 BF=4 :: RF=14 :: GOSUB 140
    0 :: Z=Z+1 :: DISPLAY AT(1
    ,8):"DATENEINGABE"
420 DISPLAY AT(3,1):"TITEL:" :
    : ACCEPT AT(4,1)SIZE(28)BE
    EP:T$(Z,1)
430 DISPLAY AT(6,1):"BAND NR:"
    :: ACCEPT AT(7,1)SIZE(28)
    BEEP:T$(Z,2)
440 DISPLAY AT(9,1):"SCHAUSPIE
    LER 1:" :: ACCEPT AT(10,1)
    SIZE(28)BEEP:T$(Z,3)
450 DISPLAY AT(12,1):"SCHAUSPI
    ELER 2:" :: ACCEPT AT(13,1
    )SIZE(28)BEEP:T$(Z,4)
460 DISPLAY AT(15,1):"SUCHWORT
     1:" :: ACCEPT AT(16,1)SIZ
    E(28)BEEP:T$(Z,5)
470 DISPLAY AT(18,1):"SUCHWORT
     2:" :: ACCEPT AT(19,1)SIZ
    E(28)BEEP:T$(Z,6)
480 DISPLAY AT(21,1):"WEITERE
    EINGABEN ))) 1" :: DISPLAY
     AT(23,1)BEEP:"EINGABE BEE
    NDET  ))) 2"
490 CALL KEY(0,K,S):: IF NOT S
     THEN 490
500 IF K<49 OR K>50 THEN 490 E
    LSE IF K=50 THEN T$(Z+1,1)
    ="ZZZ" :: GOTO 150
510 IF Z=50 THEN DISPLAY AT(12
    ,5)ERASE ALLBEEP:"SPEICHER
     VOLL" :: CALL DELAY :: GO
    TO 150 ELSE GOSUB 1420 ::
    GOTO 410
520 REM ERGAENZEN
530 Z=0
540 Z=Z+1 :: IF T$(Z,1)<>"ZZZ"
     THEN IF Z=50 THEN 150 ELS
    E 540 ELSE Z=Z-1 :: GOTO 4
    10
550 REM AENDERN
560 BF=16 :: RF=14 :: GOSUB 14
    00
570 DISPLAY AT(1,7):"DATEI AEN
    DERN" :: DISPLAY AT(7,1):"
    DRUECKEN SIE TASTE...FUER.
    .."
580 DISPLAY AT(5,1):"...1  FEH
    LER BERICHTIGEN" :: DISPLA
    Y AT(7,1):"...2  CASSETTEN
    INHALT AEND."
590 CALL KEY(0,K,S):: IF NOT S
     THEN 590 ELSE IF K<49 OR
    K>50 THEN 590
600 IF K=50 THEN 660
610 DISPLAY AT(10,7):"FEHLER B
    EI:" :: DISPLAY AT(11,1):"
    ...1  TITEL"
620 DISPLAY AT(13,1):"...2  BA
    NDNUMMER" :: DISPLAY AT(15
    ,1):"...3  SCHAUSPIELER"
630 DISPLAY AT(17,1):"...4  SU
    CHWORT" :: DISPLAY AT(19,1
    ):"DRUECKEN SIE DIE ENTSPR
    ECHEN" :: DISPLAY AT(21,1)
    :"DE TASTE."
640 CALL KEY(0,K,S):: IF NOT S
     THEN 640 ELSE IF K<49 OR
    K>52 THEN 640
650 ON K-48 GOTO 800,850,900,9
    60
660 DISPLAY AT(12,4):"WELCHE B
    ANDNR.?" :: ACCEPT AT(14,9
    ):B$
670 GOSUB 1420 :: Z=0
```

Generell wird am Ende eines Programmteils ohne besondere Anforderung wieder das Menü aufgesucht. Jeder Programmteil hat übrigens eine eigene Bildschirm- und Randfarbe. Zum Schluß noch ein Tip: Als »Suchworte« zu jedem Film haben sich das Land (zum Beispiel USA, Frankreich, BRD) und das Genre (zum Beispiel Spielfilm, Krimi, Science Fiction) des jeweiligen Films bewährt. Dadurch erhält man zum Beispiel sofort ein Verzeichnis aller Krimis, wenn im Programmteil »suchen« das Wort »Krimi« als Kriterium eingegeben wird.

(Alwin Ertl)

```
680 Z=Z+1 :: IF T$(Z,2)=B$ THE
    N 700 ELSE IF Z<50 THEN 68
    0
690 DISPLAY AT(12,6):"BAND ";B
    $ :: DISPLAY AT(13,6):"NIC
    HT GESPEICHERT." :: CALL D
    ELAY :: GOTO 150
700 DISPLAY AT(1,1):"FOLGENDE
    DATEN GEFUNDEN:"
710 FOR I=1 TO 6 :: DISPLAY AT
    (2+I,1):T$(Z,I):: NEXT I
720 DISPLAY AT(10,1):"WELCHE Z
    EILE AENDERN?" :: DISPLAY
    AT(11,1):"(7 = WEITERSUCHE
    N)"
730 ACCEPT AT(12,14)("1234567"
    )SIZE(1):A :: IF A=7 THEN
    GOSUB 1420 :: GOTO 680 ELS
    E A$=T$(Z,A)
740 DISPLAY AT(18,1):"ALTER TE
    XT:" :: DISPLAY AT(19,1):A
    $
750 DISPLAY AT(20,1):"NEUER TE
    XT:" :: ACCEPT AT(21,1)SIZ
    E(28):A$
760 T$(Z,A)=A$ :: GOSUB 1420 :
    : DISPLAY AT(10,1):"AENDER
    UNGEN ABGESCHLOSSEN?"
770 DISPLAY AT(14,1):"(1) = JA
    (2) = NEIN"
780 CALL KEY(0,K,S):: IF NOT S
    THEN 780 ELSE IF K<49 OR
    K>50 THEN 780
790 ON K-48 GOTO 150,550
800 GOSUB 1420 :: DISPLAY AT(1
    ,1):"TITEL:" :: Z=0
810 DISPLAY AT(6,1):"1=WEITER"
    :: DISPLAY AT(7,1):"2=AEN
    DERN" :: DISPLAY AT(8,1):"
    3=ENDE"
820 Z=Z+1 :: IF T$(Z,1)="ZZZ"
    OR Z>50 THEN 150 ELSE DISP
    LAY AT(3,1):T$(Z,1)
830 CALL KEY(0,K,S):: IF NOT S
    THEN 830 ELSE IF K<49 OR
    K>51 THEN 830
840 IF K=49 THEN 820 ELSE IF K
    =51 THEN 150 ELSE ACCEPT A
    T(3,1)SIZE(-28):T$(Z,1)::
    GOTO 830
850 GOSUB 1420 :: DISPLAY AT(1
    ,1):"BANDNUMMER:" :: Z=0
860 DISPLAY AT(6,1):"1=WEITER"
    :: DISPLAY AT(7,1):"2=AEN
    DERN" :: DISPLAY AT(8,1):"
    3=ENDE"
870 Z=Z+1 :: IF T$(Z,1)="ZZZ"
    OR Z>50 THEN 150 ELSE DISP
    LAY AT(3,1):T$(Z,2)
880 CALL KEY(0,K,S):: IF NOT S
    THEN 880 ELSE IF K<49 OR
    K>51 THEN 880
890 IF K=49 THEN 870 ELSE IF K
    =51 THEN 150 ELSE ACCEPT A
    T(3,1)SIZE(-28):T$(Z,2)::
    GOTO 880
900 GOSUB 1420 :: DISPLAY AT(1
    ,1):"SCHAUSPIELER:" :: DIS
    PLAY AT(6,1):"IM FILM" ::
    Z=0
910 DISPLAY AT(10,1):"1=WEITER
    ",,"2=ZEILE 1 AENDERN","3=
    ZEILE 2 AENDERN","4=ENDE
920 Z=Z+1 :: IF Z>50 OR T$(Z,1
    )="ZZZ" THEN 150 ELSE DISP
    LAY AT(3,1):T$(Z,3):: DISP
    LAY AT(4,1):T$(Z,4)
930 DISPLAY AT(8,1):T$(Z,1)
940 CALL KEY(0,K,S):: IF NOT S
    THEN 940 ELSE IF K<49 OR
    K>52 THEN 940
950 IF K=49 THEN 920 ELSE IF K
    =52 THEN 150 ELSE ACCEPT A
    T(-47+K,1)SIZE(-28):W$ ::
    T$(Z,-47+K)=W$ :: GOTO 940
960 GOSUB 1420 :: DISPLAY AT(1
    ,1):"SUCHWORTE:" :: DISPLA
    Y AT(6,1):"ZU:" :: Z=0
970 DISPLAY AT(18,1):"1=WEITER
    " :: DISPLAY AT(19,1):"2=S
    W 1 AENDERN"
980 DISPLAY AT(20,1):"3=SW 2 A
    ENDERN" :: DISPLAY AT(21,1
    ):"4=ENDE"
990 Z=Z+1 :: IF Z>50 OR T$(Z,1
    )="ZZZ" THEN 150 ELSE DISP
    LAY AT(3,1):T$(Z,5):: DISP
    LAY AT(4,1):T$(Z,6)
1000 DISPLAY AT(8,1):T$(Z,1)::
     DISPLAY AT(9,1):T$(Z,2):
     : DISPLAY AT(10,1):T$(Z,3
     )):: DISPLAY AT(11,1):T$(Z
     ,4)
1010 CALL KEY(0,K,S):: IF NOT
     S THEN 1010 ELSE IF K<49
     OR K>52 THEN 1010
1020 IF K=49 THEN 990 ELSE IF
     K=52 THEN 150 ELSE ACCEPT
     AT(-47+K,1)SIZE(-28):W$
     :: T$(Z,-45+K)=W$ :: GOTO
     1010
1030 BF=8 :: RF=13 :: GOSUB 14
     00 !SUCHE
1040 DISPLAY AT(1,10):"SUCHE"
     :: DISPLAY AT(3,1):"WAEHL
     EN SIE:"
1050 DISPLAY AT(5,1):"INHALT E
     INES BANDES ))) 1" :: DIS
     PLAY AT(7,1):"TITEL SUCHE
     N      ))) 2"
1060 DISPLAY AT(9,1):"SCHAUSPI
     ELER SUCHEN ))) 3" :: DIS
     PLAY AT(11,1):"STICHWORT
     ABFRAGEN  ))) 4"
1070 DISPLAY AT(13,1):"RUECKKE
     HR" :: DISPLAY AT(13,21):
     ")))  5" :: DISPLAY AT(15,
     9):"IHRE WAHL:"
1080 ACCEPT AT(17,13)BEEP("123
     45")SIZE(1):A :: GOSUB 14
     20 :: ON A GOTO 1090,1170
     ,1230,1290,150
1090 DISPLAY AT(1,7):"BANDINHA
     LT" :: DISPLAY AT(3,1):"W
     ELCHES BAND?" :: ACCEPT A
     T(3,15):B$ :: DISPLAY AT(
     3,1):""
1100 DISPLAY AT(1,18):B$ :: Z=
     0
1110 Z=Z+1 :: IF Z>50 OR T$(Z,
     1)="ZZZ" THEN 1390 ELSE I
     F T$(Z,2)<>B$ THEN 1110
1120 DISPLAY AT(3,1):T$(Z,1)::
      DISPLAY AT(5,1):"HIT:"
1130 DISPLAY AT(6,1):T$(Z,3)::
      DISPLAY AT(7,1):T$(Z,4)
1140 DISPLAY AT(9,1):"UNTER SU
     CHWORT:" :: DISPLAY AT(10
     ,1):T$(Z,5):: DISPLAY AT(
     11,1):T$(Z,6)
1150 DISPLAY AT(15,1)BEEP:"BIT
     TE TASTE DRUECKEN."
1160 CALL KEY(0,K,S):: IF NOT
     S THEN 1160 ELSE GOSUB 14
     20 :: GOTO 1110
1170 GOSUB 1420 :: DISPLAY AT(
     1,8):"TITELSUCHE" :: Z=0
1180 DISPLAY AT(3,1):"WELCHER
     TITEL?" :: ACCEPT AT(4,1)
     SIZE(28)BEEP:B$
1190 Z=Z+1 :: IF Z>50 OR T$(Z,
     1)="ZZZ" THEN 1390 ELSE I
     F T$(Z,1)<>B$ THEN 1190
1200 DISPLAY AT(5,1):T$(Z,1)::
      DISPLAY AT(7,1):"HIT" ::
      DISPLAY AT(8,1):T$(Z,3):
     : DISPLAY AT(9,1):T$(Z,4)
1210 DISPLAY AT(11,1):"AUF CAS
     SETTE" :: DISPLAY AT(12,1
     ):T$(Z,2):: DISPLAY AT(20
     ,1):"BITTE TASTE DRUECKEN
     ."
1220 CALL KEY(0,K,S):: IF NOT
     S THEN 1220 ELSE 150
1230 GOSUB 1420 :: DISPLAY AT(
     1,8):"SCHAUSPIELER" :: Z=
     0
1240 DISPLAY AT(3,1):"GES. SCH
     AUSPIELER EINGEBEN." :: A
     CCEPT AT(4,1)SIZE(28)BEEP
     :B$
1250 Z=Z+1 :: IF Z>50 OR T$(Z,
     1)="ZZZ" THEN 1390 ELSE I
     F T$(Z,3)<>B$ AND T$(Z,4)
     <>B$ THEN 1250
1260 DISPLAY AT(6,1):T$(Z,3)::
      DISPLAY AT(7,1):T$(Z,4):
     : DISPLAY AT(8,1):"IN" ::
      DISPLAY AT(9,1):T$(Z,1)
1270 DISPLAY AT(11,1):"AUF CAS
     SETTE" :: DISPLAY AT(12,1
     ):T$(Z,2):: GOSUB 1350
1280 IF CT THEN 1250 ELSE 150
1290 GOSUB 1420 :: DISPLAY AT(
     1,8):"SUCHWORT" :: Z=0
1300 DISPLAY AT(3,1):"SUCHWORT
      BITTE EINGEBEN." :: ACCE
     PT AT(4,1)SIZE(28)BEEP:B$
1310 Z=Z+1 :: IF Z>50 OR T$(Z,
     1)="ZZZ" THEN 1390 ELSE I
     F T$(Z,5)<>B$ AND T$(Z,6)
     <>B$ THEN 1310
1320 DISPLAY AT(6,1):"SUCHWORT
     E" :: DISPLAY AT(7,1):T$(
     Z,5):: DISPLAY AT(8,1):T$
     (Z,6):: DISPLAY AT(10,1):
     "FUER FILM"
1330 DISPLAY AT(11,1):T$(Z,1):
     : DISPLAY AT(13,1):"HIT"
     :: DISPLAY AT(14,1):T$(Z,
     3):: DISPLAY AT(15,1):T$(
     Z,4):: DISPLAY AT(17,1):"
     AUF BAND"
1340 DISPLAY AT(18,1):T$(Z,2):
     : GOSUB 1350 :: IF CT THE
     N 1310 ELSE 150
1350 CT=0 :: DISPLAY AT(23,1)B
     EEP:"1=RICHTIG" :: DISPLA
     Y AT(24,1)BEEP:"2=WEITERS
     UCHEN"
1360 CALL KEY(0,K,S):: IF NOT
     S OR K<49 OR K>50 THEN 13
     60
1370 IF K=50 THEN CT=1
1380 DISPLAY AT(23,1):"" :: DI
     SPLAY AT(24,1):"" :: RETU
     RN
1390 DISPLAY AT(10,9)ERASE ALL
     BEEP:"NICHT GEFUNDEN" ::
     CALL DELAY :: GOTO 150
1400 FOR I=1 TO 8 :: CALL COLO
     R(I,2,BF):: NEXT I
1410 CALL SCREEN(RF):: CALL CO
     LOR(0,RF,1):: CALL SOUND(
     100,-1,0)
1420 PRINT : : : : : : : : : :
      : : : : : : : : : : : :
     : : ::: RETURN
1430 SUB DELAY
1440 FOR I=1 TO 800 :: NEXT I
     :: SUBEND
1450 SUB SORT(A$(,))
1460 Z=0
1470 Z=Z+1 :: IF Z=50 OR A$(Z,
     1)="ZZZ" THEN 1480 ELSE 1
     470
1480 FOR J=1 TO Z-1 :: FOR I=J
     +1 TO Z
1490 IF A$(J,1)<=A$(I,1)THEN 1
     510
1500 FOR K=1 TO 6 :: C$=A$(I,K
     ):: A$(I,K)=A$(J,K):: A$(
     J,K)=C$ :: NEXT K
1510 NEXT I :: NEXT J :: SUBEN
     D
1520 END
```

Listing »Videothek«

OSBORNE-1DQD

inkl. FIBU, Spiele, COBOL oder FORTRAN o. PASCAL. A/D-Wandler, 3 Kanäle, 12 Bit Auflösung, Uhr, 2x8 Bit Ein-/Ausgang DM 650,—

OSBORNE EXECUTIVE

128 KRAM, CP/M 3.0, Wordstar 3.3, Personal Pearl, Fibu und Spiele

OLIVETTI-M20

ab 5980,— inkl. Spiele, Fibu, Auftragswesen, Lagerwesen, geeignet für IEEE-Bus, CP/M-86, MS-DOS

M 10

BASIC-Taschencomputer 24 KB, LCD-Anzeige

SCHOMMER COMPUTER & ELECTRONIC GMBH, Graf-Philipp-Str. 5, 6600 Saarbrücken, Tel. 0681/581832

Starke Titel für den TI 99/4A

Im Leben ist der erste Eindruck oft der entscheidende, bei Programmen nicht selten ebenso. Die Produktmanager der Software-Häuser wissen das und verwenden gerade bei Spielprogrammen viel Phantasie auf die Titel. Eigene Programme brauchen auch in dieser Beziehung keine Stiefkinder zu sein. Wir zeigen, wie man professionell wirkende Titel macht — ein Extended Basic-Modul vorausgesetzt.

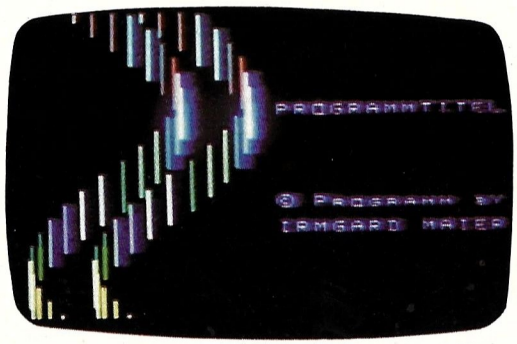
Fünf Titelroutinen für den TI 99/4A sollen zeigen, mit welchen Effekten man bei Titeln arbeiten kann, wenngleich es außer den gezeigten natürlich noch viel mehr Möglichkeiten gibt. Sehr hilfreich für gute Titelfeffekte ist die Fähigkeit des Extended Basic, sogenannte Sprites erzeugen zu können. Diese besonderen Zeichen, die bis zu viermal so groß wie ein normales Zeichen sein dürfen und frei definierbar sind, können unabhängig vom ablaufenden Programm punktweise über den Bildschirm bewegt werden. Diese Fähigkeit wird in allen gezeigten Titelbeispielen angewandt.

Am unmittelbarsten benützen wir diese Eigenschaft der Sprites in der Titelroutine »Slalom« (Listing 1). Die Routine umfaßt nur acht Zeilen (die erste und die letzte sind im Rahmen eines Hauptprogramms nicht nötig) und erzeugen eine Reihe von abwärts fließenden geschlängelten Doppelstrichen. Der Effekt ist rein dekorativ. Die Striche können aber in ihrer Form je nach Programminhalt anders definiert werden. Das wichtigste jedoch ist, daß nach der Initialisierung dieser Routine das Hauptprogramm bereits beginnen kann, während auf dem Schirm der Titeleffekt ungestört weiter abläuft. Auf diese Weise kann eine länger dauernde Anlaufphase des Hauptprogramms (wenn zum Beispiel Daten eingelesen werden) für den Anwender kaschiert werden. Das ist schöner als vor dem leeren Bildschirm warten zu müssen. Man kann sogar in das Titelbild aktuelle Texte einblenden (zum Beispiel mit einer Laufschrift). Auf diese Weise kann eine Ausgabe von Bedienungshinweisen erfolgen.

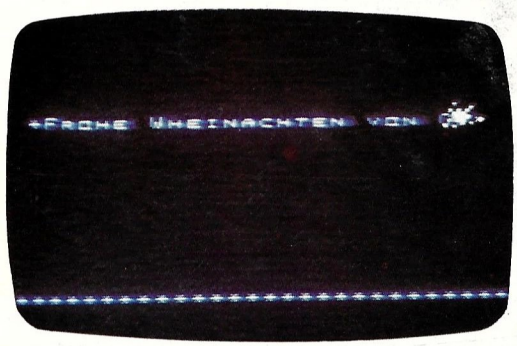
Etwas trickreicher ist die Anwendung der Sprites in der Titelroutine »Tropfende Zeichen« (Listing 2) gestaltet. Obwohl diese Routine wiederum nur wenige Zeilen umfaßt, ist ihre Wirkung verblüffend. Von der Oberkante des Bildschirms fallen Zeichen herab, die, je näher sie der eigentlichen Ausgabezeile kommen, desto langsamer werden, um schließlich scheinbar ruckfrei aufzusetzen und Zeichen für Zeichen den Ausgabetext zu bilden. Der Hintergrund kann übrigens (mit Ausnahme der Ausgabezeile) farblich und grafisch beliebig gestaltet werden, da die fallenden Zeichen (als Sprites) von ihm unabhängig sind. Im Gegensatz zum vorhergehenden Titel kann aber nebenbei kein weiteres Programm ablaufen.

Ein Titeleffekt, der recht bieder wirkt, kann mit der Routine »Cosmic Adventures« (Listing 3) erzeugt werden. Die einzige Besonderheit liegt in der Anwendung des Befehls »CALL LOAD (—31878,x)«, wobei x für die Anzahl der Sprites steht, die nach dem Subroutinenaufruf noch bewegt werden können. Mit diesem Trick werden in einer Schleife erst 26 bewegte Sprites erzeugt, anschließend alle bis auf ein Sprite mit »CALL LOAD (—31878,1)« fixiert und mit dem Befehl »CALL LOAD (—31878,26)« schlagartig alle wieder in Bewegung versetzt.

Die komplizierteste aller hier vorgestellten Titelroutinen, und fast schon ein eigenständiges Programm, ist die Routine »Stardust« (Listing 4). Dieser Titeleffekt zeigt in dreidimensionaler Darstellung die Sonne, um die ein Planet kreist. Hierbei wird abweichend von den bisherigen Routinen, eine ständige Korrektur der Spritebewegung durchgeführt, weil die automatische Spritebewegung normalerweise nur eine gradlinige Bewegung zuläßt. Im vorliegenden Bild muß sich aber der Planet elipsenförmig bewegen. Grundlage der Bahnkorrektur ist der Rechenalgorithmus in Zeile 72. Um kleine Ungenauigkeiten, die unter Umständen von Zyklus zu Zyklus auftreten (ein Nachteil der Spritetechnik



»Slalom« Das Hauptprogramm läuft bereits, während sich noch die Linien schlängeln



»Sternwerfer« weihnachtliche Stimmung vermittelt diese Titelroutine

»Stardust« Durch einen Kunstgriff entsteht der Eindruck einer dreidimensionalen Umkreisung der Sonne

»Coscic Adventure« Ein Titeleffekt, der etwas bieder wirkt, aber blitzschnelle Bewegung beinhaltet

ist mangelnde Kontrolle der Bewegung), ausgleichen zu können, wird einmal pro Kreisbahn (wenn der Planetensprite hinter der »Sonne« steht) die Position des Planeten in Zeile 72 mit CALL LOCATE exakt auf einen Punkt definiert.

Durch einen Kunstgriff wandert der Planet einmal vor und einmal hinter der Sonne vorbei

Noch ein Trick steckt in der Routine. Die Sonne besteht aus zwei übereinander liegenden Sonnensymbolen. Einmal ist eine Sonnenscheibe auf ganz normale Weise als Zeichen auf dem Bildschirm dargestellt. Exakt darüber existiert eine zweite Sonnenscheibe in Form eines fest positionierten Sprites. Wenn nun der Planet für den Betrachter hinter der Sonne vorbeifliegen soll, muß er optisch durch die Sonne verdeckt werden. Ein Sprite kann aber nur durch ein anderes Sprite mit einer niedrigeren Ordnungszahl verdeckt werden, nicht durch ein normales Zeichen. Wird Sprite Nummer 1, das Sonnensprite, aktiviert, verdeckt dieses das Sprite Nummer 2, den Planeten. Soll der Planet aber vor der Sonne vorbeifliegen, dann wird das Sonnensprite gelöscht und der Planet wandert über die aus Normalzeichen bestehende Sonnenscheibe. Die Planetenschatten, die über die Texte laufen, sind natürlich auch nur Sprites, allerdings pechschwarze. Die dreidimensionale Wirkung beruht nur auf den Perspektiven und einer sehr subtilen Abstimmung der einzelnen Bewegungsabläufe untereinander.

Nachdem wir ein so kompliziertes Programm hinter uns gebracht haben, soll noch ein kleine Titelroutine gezeigt werden, die vorzüglich in die Jahreszeit paßt: »Sternwerfer« (Listing 5). Die Buchstaben eines vorgegebenen Textes blitzen dabei wie Sternwerfer auf. Der Hintergrund könnte in Form eines farbigen Weihnachtsbaumes gestaltet werden, zu dem die Routine den Text zaubert. Hoffentlich regen diese Beispiele noch zu vielen Ideen an, sehr viele Programme könnten es gebrauchen. (Irmgard Maier)

```
50 !****TITELSCREEN: STARDUST ****
51 M$="STARDUST" ! HIER KANN DER EIGENE PROGRAMMTITEL EINGESETZT WERDEN, ER SOLL
 ABER NICHT MEHR ALS 8 ZEICHEN ENTHALTEN
52 AUT$="IRMGARD MAIER" ! HIER KANN DER EIGENE NAME EINGESETZT WERDEN, ER SOLL A
BER NICHT MEHR ALS 13 ZEICHEN ENTHALTEN
53 CALL CLEAR :: CALL SCREEN(2)
54 FOR N=5 TO 12 :: CALL COLOR(N,14,1):: NEXT N
55 RANDOMIZE
56 CALL CHAR(40,"1"):: CALL CHAR(48,"101"):: CALL CHAR(56,"00001")!STERNE
57 CALL CHAR(136,"0003DF1F3F3F7F7F7F7F3F3F1FDF030000C0FDF8FCFCFEFEFEFEFCFCF8FDC0
00")!SONNE
58 CALL CHAR(128,"0000000000000001010000000000000000000000000000808")!ERDE
59 CALL CHAR(127,"3C4299A1A199423C")
60 CALL CHAR(140,"0000000000000103030100000000000000000000000080CDC08")!ERDSCHAT
TEN
61 FOR N=1 TO 90 :: CALL VCHAR(INT(RND*23)+1,INT(RND*31)+1,8*(INT(RND*3)+5)):: N
EXT N
62 DISPLAY AT(23,1):CHR$(127)&" Programm by":AUT$ :: DISPLAY AT(3,14):M$
63 CALL COLOR(2,16,1,3,11,1,4,6,1)
64 CALL MAGNIFY(3)
65 CALL CHAR(132,"0")
66 CALL SPRITE(#2,128,4,64,120,0,0,#1,138,16,70,120,0,0,#3,138,16,70,120,0,0)
67 FOR N=1 TO 63
68 A=N/10
69 IF COS(A)>0 THEN CALL PATTERN(#1,136):: CALL COLOR(#2,4,#4,2)ELSE CALL PATTER
N(#1,132):: CALL COLOR(#2,14,#4,1)
70 IF N=31 THEN CALL SPRITE(#5,136,2,178,140,0,-50)ELSE IF N=36 THEN CALL DELSPR
ITE(#5)
71 IF N=1 THEN CALL SPRITE(#4,140,2,13,105,0,40)ELSE IF N=5 THEN CALL DELSPRITE(
#4)
72 CALL MOTION(#2,SIN(A),10*COS(A)-.1):: IF N=1 THEN CALL LOCATE(#2,64,120)
73 CALL KEY(0,ST,K):: IF ST>0 THEN 76
74 NEXT N
75 GOTO 67
76 CALL CLEAR ! HIER KANN EIGENES PROGRAMM BEGINNEN.
```

Listing 4. Fast ein eigenes Programm: Sonne und Erde aus kosmischer Sicht durch listenreichen Einsatz von Sprites

```
50 REM ***TITELSCREEN: TROPFENDE ZEICHEN***
51 Z=20 ! AUSGABEZEILE(MINIMUM=8)
52 M$="** Dies ist ein PROBETEXT **" ! AUSGABETEXT(28 ZEICHEN)
53 CALL CLEAR :: CALL SCREEN(2):: FOR N=0 TO 14 :: CALL COLOR(N,16,1):: NEXT N
54 FOR N=1 TO 28
55 CALL SPRITE(#1,32,16,1,N*8+9,100,0):: ZZ=Z*8-7
56 C=ASC(SEG$(M$,N,1)):: IF C=32 THEN 60
57 CALL PATTERN(#1,C)
58 CALL POSITION(#1,ZS,SS):: CALL MOTION(#1,(ZZ-ZS)/1.5,0):: IF ZS<ZZ THEN 58
59 CALL HCHAR(Z,N+2,C)
60 NEXT N
61 CALL CLEAR ! HIER KANN DAS EIGENE PROGRAMM BEGINNEN.
```

Listing 2. Kurz aber wirkungsvoll: exponential abgebremste Buchstaben formieren sich zum Titeltext

```
50 REM **TITELSCREEN: SLALOM  **ALLE VARIABLEN SIND INTERN ABGESCHLOSSEN
51 M$="PROGRAMMTITEL" ! HIER KANN DER EIGENE TITEL EINGETRAGEN WERDEN
52 AUT$="IRMGARD MAIER" ! HIER KANN DER EIGENE NAME EINGESETZT WERDEN
53 CALL CLEAR :: CALL SCREEN(2):: CALL MAGNIFY(4)
54 CALL CHAR(136,RPT$("80",16)&RPT$("01",16)):: CALL CHAR(127,"3C4299A1A199423C"
)
55 FOR N=5 TO 12 :: CALL COLOR(N,14,1):: NEXT N :: DISPLAY AT(15,16):CHR$(127)&"
 Programm by" :: DISPLAY AT(17,16):AUT$
56 DISPLAY AT(8,16):M$
57 FOR N=1 TO 25 :: CALL SPRITE(#N,136,1,230,60+35*SIN(N/4),0,0):: NEXT N
58 FOR N=1 TO 25 :: CALL MOTION(#N,36,0):: CALL COLOR(#N,INT(N/2)+3):: NEXT N
59 GOTO 59 ! HIER MUSS DER ANSCHLUSS DES NACHFOLGENDEN PROGRAMMS ERFOLGEN.
```

Listing 1. Mit dieser Titelroutine kann eine langwierige Programminitialisierung überbrückt werden

```
50 REM **TITELSCREEN: COSMIC ADVENTURES **ALLE VARIABLEN SIND INTERN ABGESCHLOSS
EN
51 M$="COSMIC ADVENTURES" ! HIER KANN EIGENER TITEL EINGEFUEGT WERDEN
52 AUT$="IRMGARD MAIER" ! HIER KANN EIGENER NAME EINGEFUEGT WERDEN (BIS ZU 13 ZE
ICHEN)
53 CALL CLEAR :: CALL SCREEN(2)
54 CALL CHAR(127,"3C4299A1A199423C"):: CALL CHAR(128,"000101")
55 CALL CHAR(132,"0000000101C363371F0C0810304884038080800C0C0E3E6ECF83010080C1221
C0")
56 CALL CHAR(136,"0307DF0F0F0F0F0707070703030301010E0F0F0F0E0E0C0C0C080808")
57 FOR N=5 TO 12 :: CALL COLOR(N,14,1):: NEXT N :: DISPLAY AT(15,16):CHR$(127)&"
 Programm by" :: DISPLAY AT(17,16):AUT$
58 DISPLAY AT(7,12):M$ :: DISPLAY AT(24,2):"Beginn mit beliebiger Taste"
59 CALL MAGNIFY(3):: CALL SPRITE(#1,136,1,162,81,0,0)
60 FOR N=2 TO 26 :: CALL SPRITE(#N,128,1,96,86,RND*30-15,RND*30-15):: NEXT N
61 CALL INIT :: CALL LOAD(-31878,1):: FOR N=2 TO 26 :: CALL LOCATE(#N,96,86):: C
ALL COLOR(#N,INT(N/2)+2):: NEXT N
62 CALL LOAD(-31878,26)
63 CALL COLOR(13,5,1):: DISPLAY AT(20,9):CHR$(132);CHR$(134)
64 DISPLAY AT(21,9):CHR$(133);CHR$(135)
65 CALL COLOR(#1,12):: CALL COLOR(#1,1):: FOR N=2 TO 26 :: CALL LOCATE(#N,96,86)
:: CALL KEY(3,K,R):: IF R<>0 THEN 67
66 NEXT N :: GOTO 65
67 CALL CLEAR ! HIER KANN EIGENES PROGRAMM BEGINNEN.
```

Listing 3. Start/Stoptrick für Sprites verhelfen einer Rakete zu Titelehren

```
50 REM ***TITELSCREEN: STERNWERFER ***
51 Z=10 ! AUSGABEZEILE
52 M$=" *Frohe Weihnachten von CP* " ! AUSGABETEXT(28 ZEICHEN)
53 CALL CLEAR :: CALL SCREEN(2):: FOR N=0 TO 14 :: CALL COLOR(N,5,1):: NEXT N
54 CALL CHAR(139,""):: CALL CHAR(140,"000000001"):: CALL CHAR(141,"00000281C380
8"):: CALL CHAR(142,"2002543834580AA0")
55 CALL SPRITE(#1,139,16,1,1)
56 FOR N=1 TO 28
57 C=ASC(SEG$(M$,N,1)):: IF C<32 THEN 59
58 FOR D=1 TO 60 :: NEXT D :: GOTO 63
59 CALL LOCATE(#1,Z*8-7,N*8+9)
60 CALL MAGNIFY(1):: FOR G=0 TO 2 :: CALL PATTERN(#1,G+140):: FOR D=1 TO 15 :: N
EXT D :: NEXT G
61 CALL LOCATE(#1,Z*8-13,N*8+4):: CALL MAGNIFY(2):: FOR G=0 TO 1 :: CALL PATTERN
(#1,G+140):: FOR D=1 TO 4 :: NEXT D :: NEXT G
62 CALL PATTERN(#1,142):: CALL HCHAR(Z,N+2,C):: CALL PATTERN(#1,139)
63 NEXT N
64 CALL CLEAR ! HIER KANN EIGENES PROGRAMM BEGINNEN
```

Listing 5. Paßt in die Jahreszeit: Routine für Weihnachten und Neujahr. Prosit

```
Aktienbezeichnung: Test

   34 gespeicherte Kurse vom 12.01.1983 bis zum 15.02.1983

lfd.Nr.   Kurs  lfd.Nr.   Kurs  lfd.Nr.   Kurs  lfd.Nr.   Kurs  lfd.Nr.   Kurs
   1    122.00     2    123.00     3    122.00     4    122.00     5    124.00
   6    126.00     7    125.00     8    126.00     9    122.00    10    125.00
  11    127.00    12    130.00    13    133.00    14    135.00    15    133.00
  16    132.00    17    127.00    18    123.00    19    124.00    20    125.00
  21    123.00    22    123.00    23    124.00    24    125.00    25    130.00
  26    135.00    27    134.00    28    136.00    29    137.00    30    138.00
  31    139.00    32    143.00    33    144.00    34    142.00
```

Drucker: Ausgabe der gespeicherten Kurse

die Eingabe der Kurse, außerdem das Datum des ersten und des letzten Kurses. Die laufende Eingabe wird beendet durch Betätigen der »NEW LINE«-Taste.

Grafik ausgeben:

Die Ausgabe erfolgt, je nach Wunsch, auf dem Bildschirm oder dem Drucker. Nach Auswahl und gegebenenfalls Eingabe der Druckerbreite wird vom Programm die (Kurs-)Einheit berechnet und angezeigt, die in der Grafik durch ein Symbol dargestellt wird. Diese Einheit wird berechnet unter Berücksichtigung von Höchst- und Tiefstkurs sowie der möglichen Darstellungsgröße (eingegebene Druckerbreite beziehungsweise Bildschirmbreite 64 Zeichen). Aus diesem Grund wird der angezeigte Wert oft ungerade sein. Hier besteht die Möglichkeit zur Anpassung (Erhöhung des Wertes auf maximal das Zweifache). Dadurch wird die Grafik zwar etwas schmaler und damit minimal ungenauer, aber der Wert, der durch ein Symbol dargestellt wird, ist einfacher zu interpretieren. Die Grafik beginnt am oberen Rand und wird nach unten weitergeführt. Steigende Kurse bewegen sich zum rechten Rand, fallende Kurse zum linken Rand. Die über den Drucker ausgegebene Grafik sollte daher zur einfacheren Lesbarkeit eine viertel Umdrehung gegen den Uhrzeiger gedreht werden.

Kurse ausgeben:

Die gespeicherten Kurse werden unter Angabe ihrer laufenden Nummer auf dem Bildschirm oder Drucker ausgegeben.

Datensicherung:

Die gespeicherten Kurse werden unter der angegebenen Aktienbezeichnung gesichert (Diskette) beziehungsweise, nach Vorbereiten der Kassette, auf Band geschrieben.

Kurse berichtigen:

Die gespeicherten Kurse werden unter ihrer laufenden Nummer auf dem Bildschirm angezeigt. Zu berichtigende Kurse werden neu eingegeben; richtige Kurse werden mit der »NEWLINE«-Taste übersprungen.

(Heinz Blase)

Farbangepaßte Hardcopy mit TI 99/4A und GP 700 A

Mit dem 7-Farben-Drucker GP 700 A von Seikosha ergibt sich die Möglichkeit, farbige Bildschirm-Ausgaben des TI 99/4A in ihren Grundfarben als Hardcopy festzuhalten.

TI 99/4A + GP 700 A

Da die Programmdurchführung relativ lange Zeit in Anspruch nimmt und der TI 99/4A somit blockiert wäre, beinhaltet das vorliegende Programm mit den Maschinenprogrammen HIDE und SEEK die Möglichkeit, während des Programmablaufes komplette Bildschirme einschließlich der zugehörigen Color- und Patterntables abzuspeichern und später zurückzurufen. Nach Ablauf des Hauptprogramms kann dann die Hardcopy durchgeführt werden (Beispiel im Bild). Erschwerend bei der Programmgestaltung war die Eigenheit des TI 99/4A, im Extended-Basic keinen direkten Zugriff in das für den Bildschirm verantwortliche VDP-RAM zuzulassen, so daß eine Maschinencode-Routine PEEKV ebenfalls notwendig wurde. Alle drei Maschinencode-Routinen werden durch das Unterprogramm MASCHINE zunächst in die Adressen ab >2900 geladen. In der vorliegenden Form ist das Programm für eine maximale Zwischenspeicherung von vier später auszudruckenden Schirmbildern ausgelegt (siehe Listing).

Beispiel für eine farbige Hardcopy

Bei der Farbtreue des Bildes mußten Kompromisse insofern eingegangen werden, als alle am TI verfügbaren Rot-, Blau- und Grüntöne im Druck auf die Grundfarben reduziert werden mußten. Die am Schirm vorhandene Farbe »transparent« wurde der allgemeinen Bildschirmfarbe gleichgesetzt und »weiß« ist das verwendete Papier. Beim Abspeichern mittels »CALL HIDE (n-ter Schirm, Farbcode für Schirmfarbe)« muß die Schirmfarbe deswegen angegeben werden, weil der mittels »CALL SCREEN (x)« eingegebene Wert in ein »write-only-register« geschrieben wird und damit nicht mehr »gepeekt« werden kann.

Die für die Zusammenarbeit mit einem Hauptprogramm notwendigen Hinweise sind in den REM-Zeilen 10020 bis 10040 abgelegt. Da die abgespeicherten Bildschirme beim Einlesen eines neuen Programms nicht gelöscht werden, kann das Hardcopyprogramm auch mit RUN "CS1" vom Hauptprogramm aufgerufen werden.

(Karl Hagenbuchner) ►

```
10000 ! FARBGETREUE HARDCOPY mit TI99/4A & GP 700 A
10010 ! (c) Mag. Karl Hagenbuchner, Haidfeldstr. 2, A-4050 Traun
10020 CALL INIT :: CALL MASCHINE ! Diese Zeile muß im Hauptprogramm vorm ersten
      Aufruf von CALL HIDE(n,m) stehen !
10030 ! Im Hauptprogramm können später auszudruckende Bildschirmseiten mit CALL
HIDE(n,m), dabei 1<=n<=4, abgespeichert werden,m=Schirmfarbe
10040 FOR BL=1 TO N ! N... Anzahl der gespeicherten Seiten,korrespondiert mit n
aus CALL HIDE(n,m) !
10050 GR$=CHR$(27)&"K008" :: PO$=CHR$(27)&CHR$(16)&CHR$(0)&CHR$(192)
10060 OPEN #1:"PIO.CR" :: PRINT #1:CHR$(27);"T12";CHR$(27);"E";
10070 CALL SEEK(BL,FF):: CALL SCREEN(FF):: FOR ZEILE=1 TO 24 :: P$=PO$ :: NP=192
10080 FOR SPALTE=1 TO 32 :: CALL GCHAR(ZEILE,SPALTE,CH):: IF CH<>31 THEN CALL CH
ARPAT(CH,CH$)ELSE CH$=RPT$("0",16)
10090 CALL PEEKV(2064+INT((CH-32)/8),F):: FH=F-INT(F/16)*16 :: FV=INT(F/16)
10100 RESTORE 10320 :: IF FV=0 THEN FV=FF-1
10110 READ VC1,VC2 :: IF VC1<>FV THEN 10110 ELSE A$=CHR$(20)&CHR$(VC2)
10120 RESTORE 10320 :: IF FH=0 THEN FH=FF-1
10130 READ VC1,VC2 :: IF VC1<>FH THEN 10130 ELSE B$=CHR$(20)&CHR$(VC2)
10140 FOR I=1 TO 8 :: Y(I)=0 :: NEXT I
10150 FOR SP=1 TO 8 :: H$=SEG$(CH$,2*SP-1,1):: G$=SEG$(CH$,2*SP,1)
10160 IF ASC(H$)>58 THEN H$=STR$(ASC(H$)-55)
10170 IF ASC(G$)>58 THEN G$=STR$(ASC(G$)-55)
10180 PB=VAL(H$):: I=1 :: GOSUB 10290 :: PB=VAL(G$):: GOSUB 10290 :: NEXT SP
10190 PR$,PP$="" :: IF FV<>15 THEN 10200 ELSE PR$=RPT$(CHR$(0),8):: GOTO 10210
10200 FOR I=1 TO 8 :: PR$=PR$&CHR$(Y(I)):: NEXT I
10210 IF FH<>15 THEN 10220 ELSE PP$=RPT$(CHR$(0),8):: GOTO 10230
10220 FOR I=1 TO 8 :: PP$=PP$&CHR$(255-Y(I)):: NEXT I
10230 PRINT #1:A$;P$;GR$;PR$;B$;P$;GR$;PP$;
10240 NP=NP+8 :: VP=INT(NP/256):: HP=NP-256*VP :: P$=CHR$(27)&CHR$(16)&CHR$(VP)&
CHR$(HP)
10250 NEXT SPALTE :: PRINT #1:CHR$(10):: NEXT ZEILE
10260 PRINT #1:CHR$(12):: CLOSE #1 :: NEXT BL :: CALL CHARSET
10270 DISPLAY AT(10,8)ERASE ALL:"AUSDRUCK BEENDET"
10280 END
10290 FOR J=3 TO 0 STEP -1
10300 IF PB>=2^J THEN Y(I)=Y(I)+2^(SP-1):: PB=PB-2^J
10310 I=I+1 :: NEXT J :: RETURN
10320 DATA 1,0,2,4,3,4,4,5,5,5,6,2,7,3,8,2,9,2,10,6,11,6,12,4,13,1,14,0,15,0
10330 SUB PEEKV(ADR,VALUE)
10340 CALL LOAD(15916,INT(ADR/256),ADR-INT(ADR/256)*256):: CALL LINK("PEEKV")
10350 CALL PEEK(15918,VALUE):: SUBEND
10360 SUB HIDE(N,F)
10370 CALL LOAD(15918,N,"",15933+N,F):: CALL LINK("HIDE"):: SUBEND
10380 SUB SEEK(N,FF)
10390 CALL LOAD(15918,N):: CALL LINK("SEEK"):: CALL PEEK(15933+N,FF):: SUBEND
10400 SUB MASCHINE
10410 RESTORE 10460 :: FOR I=15744 TO 15933 :: READ P :: CALL LOAD(I,P):: NEXT I
10420 CALL LOAD(8194,62,63,63,232)
10430 CALL LOAD(16360,72,73,68,69,32,32,61,128)
10440 CALL LOAD(16368,83,69,69,75,32,32,61,164)
10450 CALL LOAD(16376,80,69,69,75,86,32,62,48)
10460 DATA 194,139,6,160,61,200,0,1,0,4,6,160,61,254,4,32,32,44,6,160
10470 DATA 62,12,4,32,32,44,6,160,62,28,4,32,32,44,4,90,194,139,6,160
10480 DATA 61,200,0,1,0,4,6,160,61,254,4,32,32,36,6,160,62,12,4,32,32
10490 DATA 36,6,160,62,28,4,32,32,36,4,90,4,198,209,160,62,46,6,198,142
10500 DATA 198,17,255,142,198,21,253,2,3,253,0,2,4,255,224,2,5,254,0,2
10510 DATA 35,3,0,2,36,0,32,2,37,2,0,6,6,22,248,4,91,2,0,30,0,4,32
10520 DATA 32,52,4,192,2,1,41,0,160,67,2,2,3,0,4,91,2,0,8,0,2,1,53
10530 DATA 0,160,68,2,2,0,32,4,91,2,0,4,0,2,1,53,128,160,69,2,2,2,0
10540 DATA 4,91,0,2,3,0,192,32,62,44,4,32,32,40,216,1,62,46,4,91
10550 SUBEND
```

Listing »Farbige Hardcopys mit dem TI 99/4A«

Zeichen und Sprites definiert mit dem TI 99/4A

Möchte man sich Zeichen oder Sprites erstellen, die im normalen Zeichensatz des TI 99/4A nicht enthalten sind, so kann man sich mit dem folgenden Programm die Arbeit wesentlich vereinfachen.

TI 99/4A

Das zu konstruierende Zeichen wird in vergrößerter Form auf dem Bildschirm entworfen und dargestellt. Anschließend wird der HEX-Code des Zeichens berechnet und angezeigt. Es können maximal vier, in einem Block zusammengefaßte, Zeichen entwickelt werden.

Hardware-Konfiguration: Für dieses Programm benötigt man den TI 99/4A und das Extended-Basic Modul. Es kann aber leicht auf das reguläre TI-Basic umgeschrieben werden. Es benötigt dann aber mehr Speicherplatz und läuft nicht so schnell wie in Extended-Basic.

Bedienung: Bei der Eingabe des Programms ist darauf zu achten, daß in den Zeilen 195 bis 220 die Leerstellen und zusammengeschriebenen Worte genauso eingetippt werden wie im Listing dargestellt (ausgenommen sind die vier Leerstellen unter den Zeilennummern). Das Programm wird mit »RUN« gestartet. Nun erscheint die Frage »Wollen Sie 1 oder 4 Zeichen entwerfen?« Gibt man »1« ein, so kann nur ein Zeichen entwickelt werden (siehe Beispiel 1); gibt man »4« ein, so können vier Zeichen konstruiert werden (siehe Beispiel 2). Nun erscheint die Frage nach der Nummer des 1. Zeichens. Es kann eine Zahl von 32 bis 143 inklusive eingegeben werden. Für Sprites, welche mit »CALL MAGNIFY(3)« oder »CALL MAGNIFY(4)« vergrößert werden sollen, ist zu beachten, daß diese Zahl ganzzahlig durch 4 teilbar sein muß. Als letztes wird noch gefragt, ob eine Bedienungsanleitung erwünscht ist oder nicht. Hierauf ist mit »J« oder »N« zu antworten.

Im ersten Fall erscheint die Bedienungsanleitung, welche so lange stehen bleibt, bis irgendeine Taste gedrückt wird. Jetzt wird der Bildschirm gelöscht, der Befehlssatz aufgelistet und ein Feld aufgezeichnet, auf welchem das Zeichen mit Hilfe eines Cursors konstruiert werden kann. Der Cursor wird mit den Cursor-Tasten E (oben), X (unten), S (rechts) und D (links) bewegt. Mit der INS-Taste (2) kann an der Stelle, an der der Cursor gerade steht, ein Punkt gesetzt werden. Danach wandert der Cursor einen Punkt weiter nach rechts (außer er befindet sich schon ganz rechts).

Für die Berechnung des HEX-Codes werden ca. 8 Sekunden benötigt

Durch Drücken der DEL-Taste (1) wird der momentane Cursorpunkt gelöscht. Mit der ERASE-Taste (3) löscht man das gesamte Feld und den HEX-Code des Zeichens. Durch Betätigung ▶

Listing zu »Character definieren«

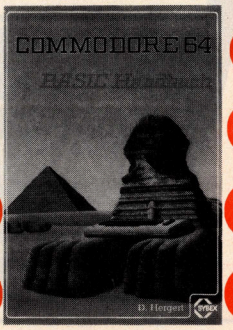
```
100 !***********************
    * Char. definieren    *
    * TI Extended BASIC   *
    * benoetigt 3704 Bytes*
    * (C) 19.10.1983 by   *
105 !* Guido Engels       *
    * Ueddinger Str. 74   *
    * 4050 Moenchengladbach 1 *
    ***********************
110 GOTO 115 :: CX :: CY :: MX :: MY :: S :: ANZ :: CHR :: ANTW$ :: I :: J :: K :: L :: DEZ :: CALL HCHAR :: CALL CHAR :: CALL KEY
115 CALL CLEAR
120 OPTION BASE 1
125 DIM FELD(16,16)
130 DIM HEX$(4)
135 !@P-  (pre-scan aus)
140 DISPLAY AT(1,2):"Char./Sprites entwickeln"
145 DISPLAY AT(3,1):"Mit diesem Programm koennen Sie Characters entwerfen."
150 DISPLAY AT(6,1):"Wollen Sie 1 oder 4 (1 Block) Zeichen entwickeln ?"
155 ACCEPT AT(7,22)SIZE(1)BEEP VALIDATE("14"):ANZ
160 DISPLAY AT(9,1):"Welche Nummer soll das Zei- chen haben ?"
165 ACCEPT AT(10,15)SIZE(3)BEEP VALIDATE(DIGIT):CHR
170 IF CHR<32 OR CHR>143 THEN 165
```


Neu von Sybex:

COMMODORE 64

BASIC Handbuch

Das COMMODORE 64 BASIC HANDBUCH zeigt Ihnen alle Anwendungsmöglichkeiten Ihres COMMODORE 64. Anhand von praktischen Beispielen wird das vollständige BASIC-Vokabular beschrieben und erläutert. Mit den vielen Beispielprogrammen lernen Sie das Programmieren mit Ihrem COMMODORE 64. Sprechen Sie die Sprache, die Ihr Computer versteht, und erleben Sie die Leistungsfähigkeit Ihres Computers.



D. Hergert
COMMODORE 64 BASIC Handbuch
ca. 184 Seiten, illustr.
Ref.-Nr. **3048** (1984)
DM 32,–

Ihr COMMODORE 64 spricht BASIC! Sprechen Sie seine Sprache!

Sybex-Bücher sind erhältlich bei Ihrem Fachändler. Fragen Sie danach!

Verlagsauslieferung:
Österreich: Fachbuch-Center ERB, Amerlingstr. 1, 1061 Wien
Schweiz: Versandbuchhandlung Thali AG, Industriestr. 2, 6285 Hitzkirch
Direktbestellungen beim Verlag gegen Verrechnungsscheck (+DM 2,50 Versandkostenanteil)

Fordern Sie ein Gesamt-Buch-Verzeichnis an.

SYBEX

SYBEX-VERLAG GMBH
Abt. CP 284 Postfach 120513
4000 DÜSSELDORF 12
Tel. 0211/287066, Telex 8588163

```
175 DISPLAY AT(12,1):"Wollen Sie eine Bedienungs- anleitung (J/N) ?"
180 ACCEPT AT(13,19)BEEP SIZE(1) VALIDATE("JN"):ANTW$
185 IF ANTW$="N" THEN 235
190 CALL CLEAR
195 DISPLAY AT(1,1):"Der Cursor wird mit den Cur-sor-Tasten E,S,D,X bewe
    gt.  Durch druecken der INS-Taste(2) wird an der Stelle an"
200 DISPLAY AT(5,1):"der der Cursor gerade steht ein Punkt gesetzt.Ansch
    lies-send wandert der Cursor ein-en Punkt weiter.Durch druecken"
205 DISPLAY AT(9,1):"der DEL-Taste(1) wird der   momentane Cursorpunkt g
    e-  loescht.Mit der ERASE-Taste (3) wird das gesammte Bild  und der
    HEX-Code geloescht."
210 DISPLAY AT(14,1):"Mit der ENTER-Taste wird derHEX-Code der Zeichen b
    erech-net und angezeigt."
215 DISPLAY AT(17,1):"Unten links erscheint das   Zeichen in Normalgroes
    se.Mitder CLEAR-Taste(4) wird das Programm beendet."
220 DISPLAY AT(23,1):"Druecken Sie eine Taste um mit dem Programm zubegi
    nnen."
225 CALL KEY(3,K,S)
230 IF S=0 THEN 225
235 DISPLAY AT(1,21)ERASE ALL:"==>  : D"
240 DISPLAY AT(3,21):"<==  : S"
245 DISPLAY AT(5,22):"^   : E"
250 DISPLAY AT(7,22):"v   : X"
255 DISPLAY AT(9,18):"DELETE  : 1"
260 DISPLAY AT(11,18):"INSERT  : 2"
265 DISPLAY AT(13,18):"ERASE   : 3"
270 DISPLAY AT(15,18):"HEX$: ENTER"
275 DISPLAY AT(17,18):"STOP    : 4"
280 CX=3 :: CY=1
285 CALL CHAR(33,"FF8181818181FF")
290 CALL CHAR(34,"FFFFFFFFFFFFFF")
295 IF ANZ=1 THEN MX=10 :: MY=8 ELSE MX=18 :: MY=16
300 FOR I=CHR TO CHR+ANZ-1
305 HEX$(I-CHR+1)="0000000000000000"
310 CALL CHAR(I,"")
315 NEXT I
320 FOR I=1 TO MY
325 CALL HCHAR(I,3,33,MX-2)
330 NEXT I
335 CALL HCHAR(21,4,CHR)
340 IF ANZ=4 THEN CALL HCHAR(21,5,CHR+1):: CALL HCHAR(22,4,CHR+2):: CALL
     HCHAR(22,5,CHR+3)
350 DISPLAY AT(21+I,5):USING "###":CHR+I
355 DISPLAY AT(21+I,8):":";HEX$(I+1)
360 NEXT I
365 CALL HCHAR(CY,CX,30)
370 CALL KEY(0,K,S)
375 IF FELD(CY,CX-2)=1 THEN CALL HCHAR(CY,CX,33)ELSE CALL HCHAR(CY,CX,33
    )
380 IF S=0 OR K<>69 AND K<>88 AND K<>83 AND K<>68 AND K<>52 AND K<>49 AN
    D K<>50 AND K<>51 AND K<>13 THEN 365
385 IF K=69 AND CY>1 THEN CY=CY-1 :: GOTO 365
390 IF K=88 AND CY<MY THEN CY=CY+1 :: GOTO 365
395 IF K=83 AND CX>3 THEN CX=CX-1 :: GOTO 365
400 IF K=68 AND CX<MX THEN CX=CX+1 :: GOTO 365
405 IF K=52 THEN CALL CLEAR :: STOP
410 IF K=49 THEN FELD(CY,CX-2)=0 :: CALL HCHAR(CY,CX,33):: IF CX<MX THEN
     CX=CX+1
415 IF K=50 THEN FELD(CY,CX-2)=1 :: CALL HCHAR(CY,CX,34):: IF CX<MX THEN
     CX=CX+1
420 IF K=49 OR K=50 THEN 365
425 IF K<>51 THEN 490
430 FOR I=1 TO MY
435 CALL HCHAR(I,3,33,MX-2)
440 FOR J=1 TO MX-2
445 FELD(I,J)=0
450 NEXT J
455 NEXT I
460 FOR I=0 TO ANZ-1
465 CALL CHAR(CHR+I,"")
470 DISPLAY AT(21+I,9):"0000000000000000"
475 NEXT I
480 CX=3 :: CY=1
485 GOTO 365
490 IF K<>13 THEN 365
495 FOR I=1 TO ANZ :: HEX$(I)="" :: NEXT I
500 FOR I=0 TO SGN(ANZ-1)
505 FOR J=0 TO SGN(ANZ-1)
510 FOR K=1 TO 16
515 DEZ=0
520 FOR L=1 TO 4
525 DEZ=DEZ+2^(4-L)*FELD(INT(.5+K/2)+I*8,4-4*(K-2*INT(K/2))+L+J*8)
530 NEXT L
535 IF DEZ<10 THEN HEX$(2*I+J+1)=HEX$(2*I+J+1)&STR$(DEZ)ELSE HEX$(2*I+J+
    1)=HEX$(2*I+J+1)&CHR$(DEZ+55)
540 NEXT K
545 NEXT J
550 NEXT I
555 FOR I=0 TO ANZ-1
560 DISPLAY AT(21+I,9):HEX$(I+1)
565 CALL CHAR(CHR+I,HEX$(I+1))
570 NEXT I
575 GOTO 365
```

Listing »Character definieren (Schluß)

der Taste »ENTER« wird der HEX-Code, der bis dahin eingegebenen Punkte berechnet. Diese Berechnung dauert pro Zeichen ungefähr acht Sekunden (bei vier Zeichen also zirka 32 Sekunden). Diese Zeit erscheint zwar recht lang, doch ist dies immer noch schneller, als wenn man selber den HEX-Code bestimmen würde. Außerdem vermeidet man mit diesem Programm Fehler, die bei einer manuellen Berechnung nicht auszuschließen sind. Nach der Berechnung des HEX-Codes, wird dieser angezeigt und unten links erscheint das Zeichen in Normalgröße. Durch Drücken der »CLEAR«-Taste wird das Programm abgebrochen.

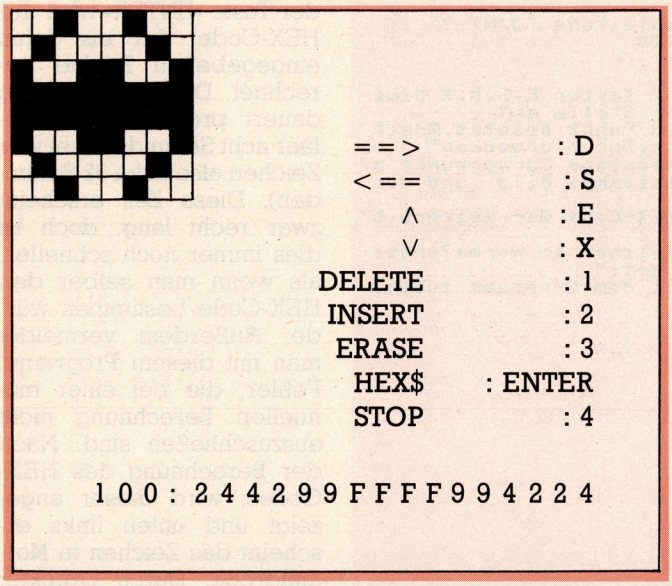
Durch Betätigung der »CLEAR«-Taste wird auch der Bildschirminhalt gelöscht

Doch Vorsicht, der Bildschirm wird auch gelöscht. Man sollte sich deswegen den HEX-Code zur Sicherheit vorher aufschreiben oder aus Zeile 405 den »CALL-CLEAR-«Befehl löschen. Sollte die »CLEAR«-Taste dennoch einmal gedrückt worden sein, bevor der HEX-Code aufgeschrieben werden konnte, so kann man unmittelbar danach durch »FOR I=1 TO 4 :: PRINT HEX$(I) :: NEXT I« diesen wieder sichtbar machen.

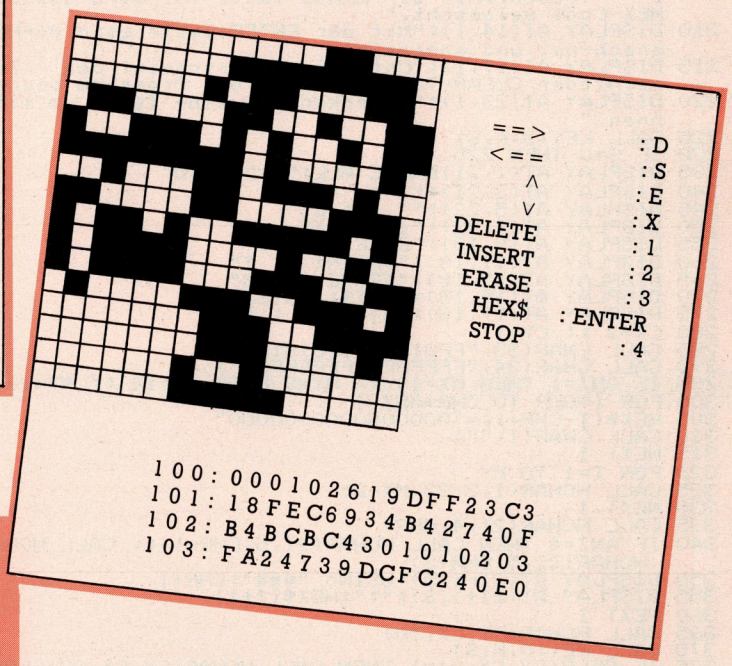
Die Tasten E, S, D und X steuern den Cusor

Programmbeschreibung: Nach der Eingabe der Anzahl der zu entwickelnden Zeichen, der Nummer (ASCII-Code) des Zeichens und der Antwort auf die Frage, ob eine Bedienungsanleitung gewünscht wird oder nicht (bei »J« wird sie ausgedruckt), gelangt man ins Haupt-Programm zurück. Die Zeilen 235 bis 360 zeichnen das vergrößerte

Warum warten?
MICROBUFFER.



Beispiel 1. Bildschirmaufbau zum Definieren eines Zeichens



Beispiel 2. Bildschirmaufbau zum Definieren von 4 Zeichen

Variable	Erklärung
CX	momentane Cursorposition in X-Richtung
CY	momentane Cursorposition in Y-Richtung
MX	maximale Cursorposition in X-Richtung
MY	maximale Cursorposition in Y-Richtung
K	Rückgabevariable für CALL KEY (bei Hex-Code Berechnung auch Laufvariable)
S	Statusvariable für CALL KEY
ANZ	Anzahl der zu entwickelnden Zeichen (1 oder 4)
CHR	Nummer des 1. zu entwerfenden Zeichens (32-143)
I	Laufvariable für FOR-Schleife
J	Laufvariable für FOR-Schleife
L	Laufvariable für FOR-Schleife
DEZ	dezimaler Wert des Zeichens
Feld (16,16)	gibt an, ob ein Punkt an einer bestimmten Stelle gesetzt ist oder nicht (Wert 1 oder 0)
HEX$(4)	Hexadezimaler Wert der Zeichen
ANTW$	Antwort ob Bedienungsanleitung gewünscht wird oder nicht ("J" oder "N")

Variablenliste

Feld eines Characters, den Befehlssatz und den (die) HEX-Code(s) der zu entwickelnden Zeichen (im ersten Durchlauf noch »0000000000000000«) auf. Die Zeilen 365 bis 380 lassen einen Cursor blinken und fragen die Tastatur nach einer gedrückten Taste ab. Wird eine Taste des Befehlssatzes gedrückt, so zweigt das Programm aus dieser Schleife heraus und führt den entsprechenden Befehl aus. Die Zeilen 385 bis 400 übernehmen die Steuerung des Cursors (Tasten E,S,D,X). Zeile 405 löscht den Bildschirm und bricht das Programm ab (Taste 4). Die Zeilen 410 bis 420 löschen oder setzen einen Punkt (Tasten 1 und 2). Die Zeilen 430 bis 485 löschen das gesamte Zeichen, setzen den Hex-Code zurück und befördern den Cursor zurück in die obere linke Ecke des Zeichens. Die Zeilen 495 bis 570 berechnen den Hexadezimal-Code der zur Definition des neuen Zeichens in der CALL CHAR-Routine notwendig ist.

Nach jeder Durchführung eines Befehls (außer Programmende) springt das Programm zurück zur Tastaturabfrage bis wieder eine Taste des Befehlssatzes gedrückt wird.

(Guido Engels)

Computer? Osborne.

Bevor Sie sich für einen Personal-Computer entscheiden, sprechen Sie besser mit unserem System-Berater: 089/40 54 26. Oder schreiben Sie an OSBORNE COMPUTER CORP. GMBH, Dingolfinger Straße 6, 8000 München 80.

OSBORNE
Management by Computer.

Der TI 99/4A als Telefonregister

Das Programm »Telefonregister« bietet dem Besitzer des TI 99/4A mit Extended Basic, auch bei nicht vorhandenem Diskettenlaufwerk, eine alternative Lösung zur Datenspeicherung.

Die Daten werden innerhalb des Programms in DATA-Zeilen gespeichert. Es entfällt somit die äußerst langwierige Suche des Computers in einer externen Datei auf Kassette. Insgesamt 99 Rufnummern stehen im vorliegenden Programm maximal auf Abruf bereit. Zu Beginn des Programms erscheint das Menü in weißer Schrift auf schwarzem Bildschirm. Es kann zwischen der »Registerbenutzung«, der »Registererweiterung«, der »Registeränderung« und der »Beendigung des Programms« gewählt werden.

Registerbenutzung:

(Zeile 31 bis 50) Der Computer fragt zuerst nach dem Namen, dann nach dem Vornamen des gesuchten Teilnehmers. Beim Familiennamen werden nur Großbuchstaben und das Zeichen »★«, beim Vornamen Großbuchstaben und die Zeichen »★ < > .« akzeptiert. Wenn Name und Vorname des Teilnehmers unbekannt (Eingabe ist dann: ★) sind, man also zu einer bestimmten Rufnummer den Teilnehmer sucht, fragt der Computer diese Rufnummer ab (hier werden nur Zahlen und das Zeichen » — « akzeptiert).

(Zeile 51 bis 69) Danach beginnt der Computer mit der Suche und gibt

(Zeile 71 bis 84) nach wenigen Sekunden Auskunft über Name, Vorname und Rufnummer des gesuchten Teilnehmers. Wird nur der Familienname eingegeben, so sucht

(Zeile 158 bis 178) der Computer alle Teilnehmer mit diesem Namen heraus und fragt dann den Vornamen ab.

Registererweiterung:

(Zeile 86 bis 114): Der Computer teilt mit:

1. Wieviele Rufnummern gespeichert sind.
2. In welcher Programmzeile die letzte Rufnummer gespeichert ist.
3. Ab welcher Zeilennummer eine Erweiterung möglich ist.
4. Wieviele neue Rufnummern noch gespeichert werden können.

Alle Eingaben müssen als Großbuchstaben erfolgen

Der Druck auf die »ENTER«-Taste führt wieder zum Menü zurück. Um eine Erweiterung vorzunehmen, muß das Programm mit »FCTN-4« unterbrochen werden. Danach die Daten des neuen Teilnehmers wie folgt eingegeben:

Zeilennummer Data "Name Vorname Vorwahlnr.1-Rufnummer + Vorwahlnr.2-Rufnummer2", Zeilennummer

Anstelle des Vornamens kann auch ein Kennwort, zum Beispiel »DR. MED.« eingegeben werden. Alle Buchstaben müssen als Großbuchstaben eingegeben werden, da sie sonst nicht vom Computer akzeptiert werden. Bei der Suche akzeptiert der Computer auch eine zweite gespeicherte Rufnummer, wenn diese von der ersten durch ein »+« getrennt ist.

Registeränderung:

(Zeile 117 bis 126) Zuerst fragt der Computer nach dem Namen und dem Vornamen des Teilnehmers.

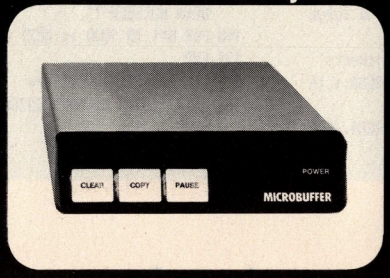
(Zeile 141 bis 154) Ist der Teilnehmer in der Datei gespeichert, so erfolgt Mitteilung über:

1. Die Zeilennummer, unter der der Teilnehmer gespeichert ist.
2. Name, Vorname und gespeicherte Rufnummern, also über den gesamten Inhalt der DATA-Zeile.

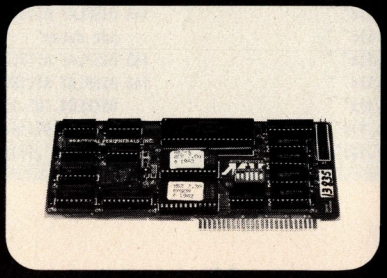
F,G,I,J,P,T,X,Z	=	Zählvariablen für FOR-NEXT-Schleifen
K	=	Rückmeldevariable im KEY-Unterprogramm
S	=	Statusvariable im KEY-Unterprogramm
LN	=	Variable für die Länge von N$
LVN	=	Variable für die Länge von VN$
LR	=	Variable für die Länge von R$
KEN KNZ	=	Variable für die Zeilennummer als Daten in der Data-Zeile
N$	=	String-Variable für den Familiennamen
VN$	=	String-Variable für den Vornamen
R$	=	String-Variable für die Rufnummer
REG$	=	String-Variable für die gesamten Daten der Data-Zeile innerhalb " "

Variablenliste zum Programm »Telefonregister«

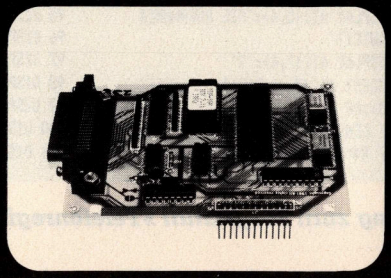
MICROBUFFER.
Die Buffer für fast jede Computer/Drucker- oder Computer/Modem-Kombination.



MICROBUFFER INLINE
universell einsetzbar.



MICROBUFFER II
speziell für Apple II/IIe.



MICROBUFFER E
für alle Epson-Drucker.

Exclusiv-Distributoren für Deutschland:

ACC Computer Handels GmbH
Rödelheimer Straße 44
6236 Eschborn/Ts.
☎ 0 61 96 - 4 30 91

EVERYWARE COMPUTERS GmbH
Postfach 22 06
6200 Wiesbaden
☎ 0 61 21 - 40 76 11

SSS-SPECIAL SOFTWARE SERVICE GmbH
An der Eisenbahn 9
6392 Neu-Anspach 1
☎ 0 60 81 - 86 38

```
1 REM**********************
2 REM  TELEFONREGISTER
3 REM  COPYRIGHT 1983 BY
4 REM  MICHAEL DONIX
5 REM  MEROWINGER STR.7
6 REM  5216 NIEDERKASSEL 6
7 REM**********************
8 CALL CLEAR :: CALL SCREEN(2)
9 FOR I=0 TO 12 :: CALL COLOR(I,16,
  1):: NEXT I
10 CALL CHAR(126,"3C4299A1A199423C"
  )
11 DISPLAY AT(7,3):"**************
  ***********"
12 DISPLAY AT(9,8):"TELEFONREGISTER
  "
13 DISPLAY AT(11,11):"ß 1983 BY"
14 DISPLAY AT(13,9):"MICHAEL DONIX"
15 DISPLAY AT(15,3):"**************
  ***********"
16 FOR F=1 TO 1000 :: NEXT F
17 REM***MENUE***
18 CALL CLEAR
19 DISPLAY AT(5,4)ERASE ALL:"TELEFO
  NREGISTER"
20 DISPLAY AT(6,4):"---------------
  "
21 DISPLAY AT(9,4):"1. REGISTERBEHU
  TZUNG"
22 DISPLAY AT(13,4):"2. REGISTERERW
  EITERUNG"
23 DISPLAY AT(17,4):"3. REGISTERAEN
  DERUNG"
24 DISPLAY AT(21,4):"4. ENDE DES PR
  OGRAMMS"
25 CALL KEY(1,K,S)
26 IF K=19 THEN 31
27 IF K=7 THEN 86
28 IF K=8 THEN 117
29 IF K=9 THEN 183 ELSE 25
30 REM***ANFRAGE***
31 DISPLAY AT(1,4)ERASE ALL:"ANFRAG
  E"
32 DISPLAY AT(2,4):"--------"
33 DISPLAY AT(22,2):"nach jeder ein
  gabe <ENTER>"
34 DISPLAY AT(5,4):"DER TEILNEHMER
  HEISST?"
35 DISPLAY AT(6,4):"(wenn unbekannt
   (*))"
36 DISPLAY AT(9,4):"NAME: ?"
37 ACCEPT AT(9,10)(UALPHA,"*")SIZE(
  -15):N$
38 DISPLAY AT(12,4):"VORNAME: ?"
39 ACCEPT AT(12,13)(UALPHA,"-.*()")
  SIZE(-15):VN$
40 IF N$="?" AND VN$="?" THEN 31 EL
  SE 41
41 LN=LEN(N$)
42 LVN=LEN(VN$)
43 IF N$="*" AND VN$="*" THEN 45 EL
  SE 44
44 IF LN<3 THEN 31 ELSE 60
45 DISPLAY AT(15,4):"DIE RUFNUMMER
  LAUTET?"
46 DISPLAY AT(17,4):"?"
47 ACCEPT AT(17,4)(DIGIT,"-")SIZE(-
  20):R$
48 IF LEN(R$)<6 THEN 45
49 IF R$="?" OR R$="*" THEN 31
50 LR=LEN(R$)
51 REM***TEILNEHMERSUCHE***
52 DISPLAY AT(11,8)ERASE ALLBEEP:"B
  ITTE WARTEN !"
53 RESTORE
54 FOR I=1 TO 500
55 READ REG$
56 IF SEG$(R$,2,LR-1)=SEG$(REG$,(PO
  S(REG$,"0",1)+1),LR-1)THEN 71
57 IF SEG$(R$,2,LR-1)=SEG$(REG$,(PO
  S(REG$,"+",1)+2),LR-1)THEN 71
58 IF REG$="ENDE" THEN 68
59 NEXT I
60 IF VN$="*" OR VN$="?" THEN 158 E
  LSE 61
61 DISPLAY AT(11,8)ERASE ALLBEEP:"B
  ITTE WARTEN !"
62 RESTORE
63 FOR J=1 TO 500
64 READ REG$
65 IF N$=SEG$(REG$,1,LN)AND (VN$=SE
  G$(REG$,LN+2,LVN)XOR VN$="*")THE
  N 71
66 IF REG$="ENDE" THEN 68
67 NEXT J
68 DISPLAY AT(9,3)BEEPERASE ALL:"KE
  INE DATEN GESPEICHERT!"
69 GOTO 80
70 REM***AUSKUNFT***
71 DISPLAY AT(1,4)BEEPERASE ALL:"AU
  SKUNFT"
72 DISPLAY AT(2,4):"--------"
73 DISPLAY AT(5,4):"DER TEILNEHMER
  HEISST:"
74 DISPLAY AT(9,4):"NAME:"
75 DISPLAY AT(9,10):SEG$(REG$,1,POS
  (REG$," ",1))
76 DISPLAY AT(12,4):"VORNAME:"
77 DISPLAY AT(12,13):SEG$(REG$,POS(
  REG$," ",1)+1,(POS(REG$,"0",1)-1
  )-(POS(REG$," ",1)))
78 DISPLAY AT(15,4):"DIE RUFNUMMER
  LAUTET:"
79 DISPLAY AT(17,4):SEG$(REG$,POS(R
  EG$,"0",1),30)
80 DISPLAY AT(22,5):"neue anfrage m
  it <1>"
81 DISPLAY AT(23,8):"index mit <2>"
82 CALL KEY(1,K,S)
83 IF K=19 THEN 31
84 IF K=7 THEN 18 ELSE 82
85 REM**DATEIERWEITERUNG**
86 CALL CLEAR
87 RESTORE
88 FOR Z=1 TO 500
89 KNZ=KEN :: IF KEN=0 THEN KNZ=490
90 READ REG$,KEN
91 PRINT TAB(2);Z;TAB(6);")";TAB(9)
  ;"ZEILENNUMMER ";KEN
92 IF REG$="ENDE" THEN 94
93 NEXT Z
94 PRINT : : : : : : : : : : :
95 DISPLAY AT(1,1):" "
96 DISPLAY AT(2,1):" "
97 DISPLAY AT(3,1):" "
98 DISPLAY AT(4,1):" "
99 DISPLAY AT(5,1):" "
100 DISPLAY AT(6,1):" "
101 DISPLAY AT(7,1):" "
102 DISPLAY AT(8,1):" "
103 DISPLAY AT(9,1):" datenbank be
  legt bis:"
104 DISPLAY AT(10,1):" "
105 DISPLAY AT(11,1):" "
106 DISPLAY AT(13,1):" "
107 DISPLAY AT(16,3):"zeile 1500 bi
  ldet das ende"
108 DISPLAY AT(17,3):"der datenbank
  ."
109 DISPLAY AT(19,3):"erweiterung d
  er datenbank"
110 DISPLAY AT(20,3):"gem. handbuch
   ab zeilen-"
111 DISPLAY AT(21,3):"nummer";KNZ+1
  0;"."
112 DISPLAY AT(23,3):"erweiterung u
  m";100-Z;"weitere"
113 DISPLAY AT(24,3):"rufnummern mo
  eglich."
114 GOTO 179
115 REM***DATEIAENDERUNG***
116 REM***TEILN.EINGABE***
117 DISPLAY AT(1,4)ERASE ALL:"DATEN
  AENDERUNG"
118 DISPLAY AT(2,4):"---------------
  "
119 DISPLAY AT(5,4):"DER TEILNEHMER
   HEISST?"
120 DISPLAY AT(23,2):"nach jeder ei
  ngabe <ENTER>"
121 DISPLAY AT(9,4):"NAME:"
122 ACCEPT AT(9,10)(UALPHA):N$
123 DISPLAY AT(12,4):"VORNAME:"
124 ACCEPT AT(12,13)(UALPHA,"().."):
  VN$
125 LN=LEN(N$)
126 LVN=LEN(VN$)
127 REM***DATEIAENDERUNG***
128 REM***TEILN.SUCHE***
129 DISPLAY AT(11,8)ERASE ALLBEEP:"
  BITTE WARTEN !"
130 RESTORE
131 FOR X=1 TO 500
132 READ REG$,KEN
133 IF N$=SEG$(REG$,1,LN)AND VN$=SE
  G$(REG$,LN+2,LVN)THEN 141
134 IF REG$="ENDE" THEN 136
135 NEXT X
136 DISPLAY AT(11,4)BEEPERASE ALL:"
  NAME NICHT GESPEICHERT!"
137 DISPLAY AT(23,7):"index mit <EN
  TER>"
138 GOTO 179
139 REM***DATEIAENDERUNG***
140 REM***TEILN.AUSKUNFT***
141 DISPLAY AT(4,2)BEEPERASE ALL:"D
  ATENAENDERUNG"
142 DISPLAY AT(5,2):"---------------
  "
143 DISPLAY AT(7,2):"unter der zeil
  ennummer"
144 DISPLAY AT(8,1):KEN;"sind folge
  nde daten"
145 DISPLAY AT(9,2):"gespeichert:"
146 DISPLAY AT(13,2):SEG$(REG$,1,(P
  OS(REG$,"0",1)-1))
147 DISPLAY AT(14,2):SEG$(REG$,(POS
  (REG$,"0",1)),28)
148 DISPLAY AT(11,1):"**************
  ***************"
149 DISPLAY AT(12,1):"*
                    *"
150 DISPLAY AT(15,1):"*
                    *"
151 DISPLAY AT(16,1):"**************
  ***************"
152 DISPLAY AT(19,2):"aenderungen i
  n der daten-"
153 DISPLAY AT(20,2):"bank gem. han
  dbuch."
154 GOTO 179
155 REM***AUFLISTUNG VON***
156 REM***TEILNEHMERN MIT**
157 REM***GLEICHEN NAMEN***
158 DISPLAY AT(11,8)ERASE ALLBEEP:"
  BITTE WARTEN !"
159 RESTORE
160 FOR P=1 TO 500
161 READ REG$
162 IF N$=SEG$(REG$,1,LN)THEN 164
163 IF REG$="ENDE" THEN 68 ELSE 166
164 CALL CLEAR :: PRINT "folgende t
  eilnehmer mit dem namen ";N$;"
  sind in"
165 PRINT "der datei gespeichert:"
  :: PRINT :: PRINT SEG$(REG$,1,(
  POS(REG$,"0",1)-1)):: GOTO 167
166 NEXT P
167 FOR T=1 TO 500
168 READ REG$
169 IF N$=SEG$(REG$,1,LN)THEN 171
170 IF REG$="ENDE" THEN 173 ELSE 17
  2
171 PRINT SEG$(REG$,1,(POS(REG$,"0"
  ,1)-1))
172 NEXT T
173 PRINT : : : : :
174 DISPLAY AT(20,1)BEEP:"gewuensch
  ter teilnehmer ?"
175 DISPLAY AT(22,1):"NAME: ";N$
176 DISPLAY AT(23,1):"VORNAME:"
177 ACCEPT AT(23,10)(UALPHA,"-","."
  ,"(",")"):VN$
178 GOTO 40
179 CALL KEY(0,K,S)
180 IF K<>13 THEN 179 ELSE 181
181 GOTO 18
182 REM***PROGRAMMENDE***
183 DISPLAY AT(4,10)ERASE ALL:"ACHT
  UNG!"
184 DISPLAY AT(5,10):"_______"
185 DISPLAY AT(7,1):"haben sie die
  datei erwei-":"tert?"
186 DISPLAY AT(10,1):"haben sie dat
  en geaendert?"
187 DISPLAY AT(15,1):"DANN BEACHTEN
   SIE, DASS SIE"
188 DISPLAY AT(17,1):"DAS PROGRAMM
  NEU AUF KAS-"
189 DISPLAY AT(19,1):"SETTE ABSPEIC
  HERN MUESSEN !"
190 FOR G=1 TO 1000 :: NEXT G
191 END
192 REM***INTERNE DATEI***
500 DATA "DONIX MICHAEL 02208-3666+
  0228-773125",500
1500 DATA "ENDE",1500
```

Listing zum Programm »Telefonregister«

Abspeichern auf Band muß nach Änderungen der Daten erfolgen

Der Druck auf die ENTER-Taste führt wieder zum Menü zurück. Um eine Änderung vorzunehmen, muß das Programm mit »FCTN-4« unterbrochen werden. Mit »Zeilennummer 'FCTN-X'« wird der Edit-Modus aufgerufen und die Änderung vorgenommen.

Beendigung des Programms:

(Zeile 183 bis 191) Wird bei der Menüwahl die Taste »4« betätigt, so erscheint vor Beendigung des Programms als Erinnerung der Hinweis, daß das Programm neu auf Kassette gespeichert werden muß, wenn während der Benutzung Änderungen oder Ergänzungen vorgenommen wurden.

Interne Datei:

Für die interne Datei wurden die Programmzeilen 500 bis 1490 (in 10er-Schritten) vorgesehen, wobei die DATA-Zeile 1500 das Ende der Datei bildet.

(Michael Donix)

Schwimmwettkampf

Hätten Sie's gewußt? Insgesamt gibt es etwa 1000 Friesen in Berlin. Am liebsten halten sie sich im Wasser auf — auch die Senioren. Als Senior wird jeder bezeichnet, der älter als 20 Jahre ist. Bevor sie jedoch zu Hunderten ins Wasser gehen können, bedarf es einiger Vorbereitungen. Und dabei leistet der TI 99/4A gute Dienste.

Bei den Berliner Friesen handelt es sich nicht etwa um ehemaliger Bewohner der gleichnamigen deutschen Landschaft, sondern um einen Schwimmverein. Zwei- bis viermal pro Woche treffen sie sich in kleineren Gruppen von 13 bis 16 Leuten, um zu trainieren. Das macht nicht nur Spaß, sondern dient auch zur Vorbereitung auf den »Ernstfall« — den nächsten Schwimmwettkampf.

Vor jedem Wettkampf gibt es eine Ausschreibung, in der die Teilnahmeberechtigung, einige allgemeine Hinweise sowie die Zusammenstellung der Wertungsgruppen und die Schwimmarten festgelegt sind. Bei dem diesjährigen sogenannten »Frühlingsschwimmen« dürfen alle Friesen-Mitglieder mitmachen, die keine Startberechtigung für einen anderen Schwimmverein haben. Gewetteifert wird nicht nur beim Brust- und Rückenschwimmen, sondern auch beim Schmetterlings- und Freistil. Die Auswertung erfolgt nach neun altersabhängigen Wertungsgruppen. Bei jedem der vier Wettkämpfe erhalten die drei Besten einer Altersgruppe jeweils eine Urkunde.

Ungefähr 500 Meldungen registriert der Koordinator, der sportliche Leiter, pro ausgeschriebenem Schwimmwettkampf. Die Anmeldungsformalitäten dafür übernimmt der Übungsleiter einer Trainingsgruppe. Er füllt für jeden, der mitmachen will, eine Startkarte (Bild 1) aus. Anhand dieser Informationen ordnet der sportliche Leiter jeden Teilnehmer den entsprechenden Wettkampfgruppen zu. Eingeteilt wird nach Alter, Geschlecht und Leistung (Bestzeit), wobei darauf geachtet wird, daß die Schwimmer einer Gruppe in etwa gleich stark sind. Beim Frühlingsschwimmen gibt es beispielsweise 16 Gruppen. Es ist klar, daß sich ein solcher Wettkampf über mehrere Tage (Abende) erstreckt. Und es ist wiederum die Aufgabe des jeweiligen Übungsleiters, jeden in seiner Trainingsgruppe darüber zu unterrichten, wann und in welcher Wettkampfgruppe er startet.

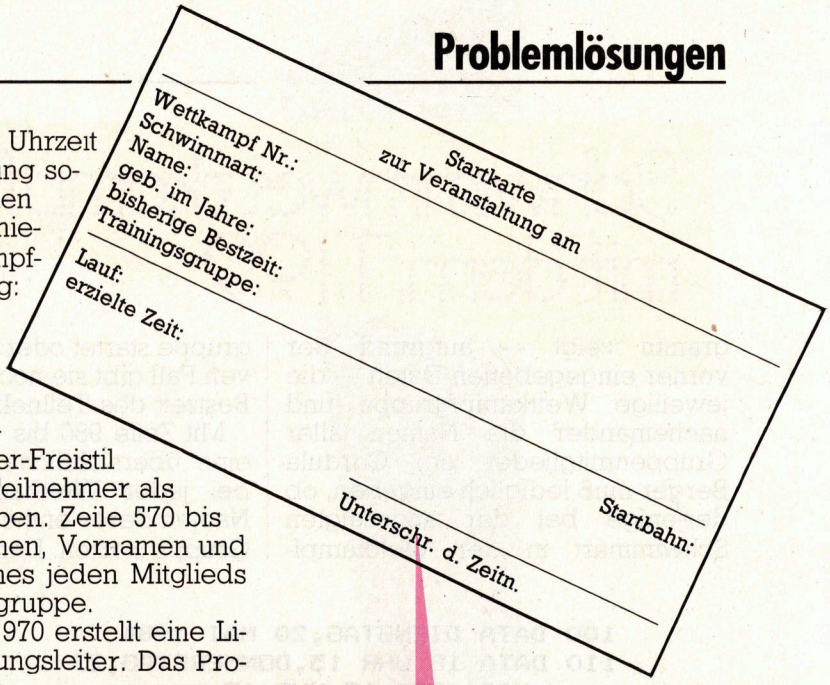
Die »Friesin« Cordula Berger, von Beruf Lehrerin, erteilt nicht nur während der offiziellen Schulzeit

Unterricht: Auch sie gehört zu denen, die in ihrer Freizeit eine Gruppe trainieren. Da ihr die Arbeit vor einem Schwimmwettkampf jedesmal von neuem lästig war, ersann sie einen Ausweg. Heute erleichtert sie sich den notwendigen »Verwaltungskram« für die ordnungsgemäßen Anmeldungen mit einem TI 99/4A. Das Programm erstellte sie selbst.

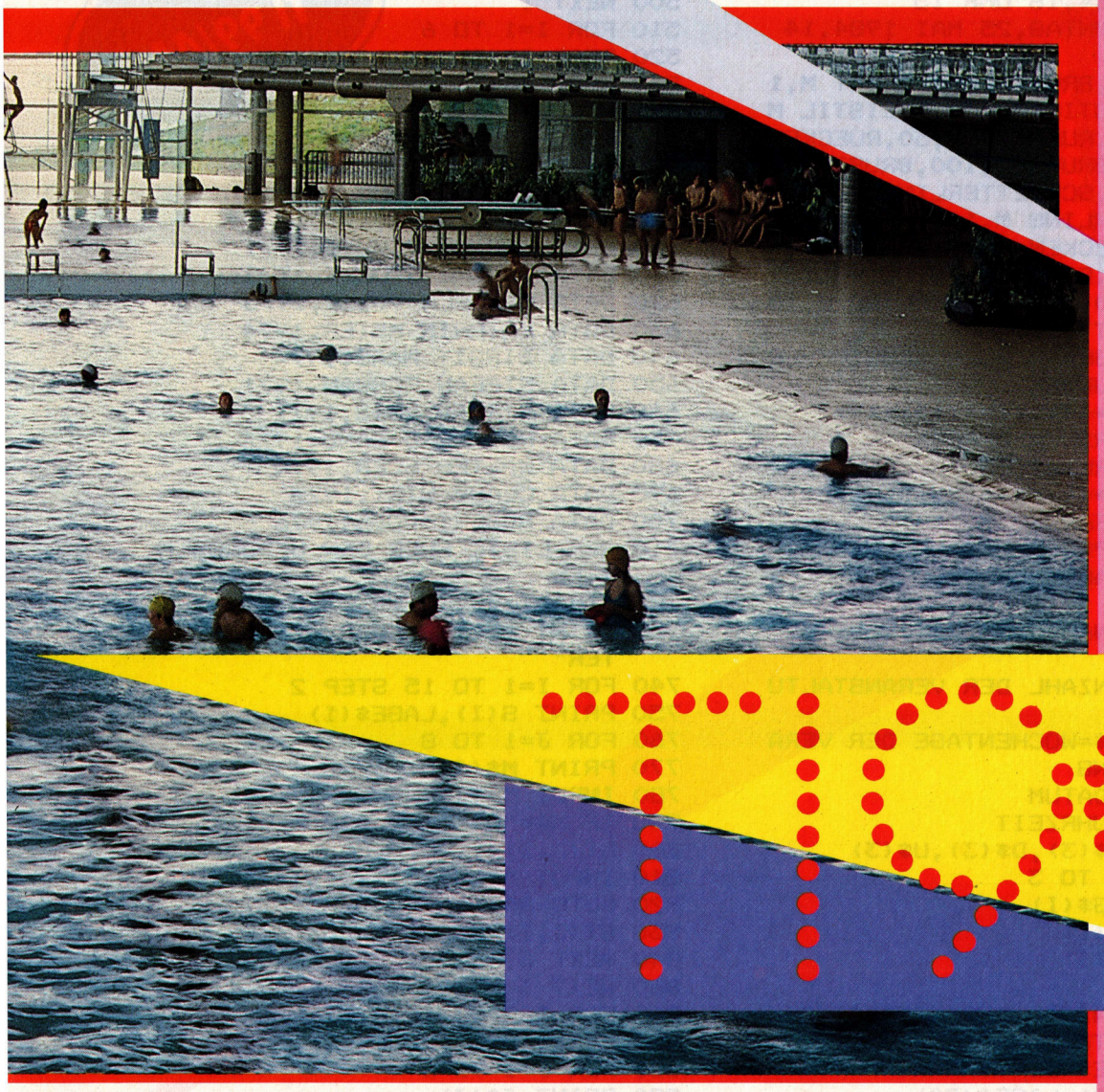
Die Verwaltungsarbeit hat der Übungsleiter

Vor einem Schwimmwettkampf gibt sie zunächst die Daten der Ausschreibung ein. Hierzu gehören Datum und Uhrzeit der Veranstaltung sowie Informationen über die verschiedenen Wettkampfgruppen (Listing: Zeile 100 bis 560). In Zeile 160 ist beispielsweise die Wettkampfgruppe 50-Meter-Freistil für weibliche Teilnehmer als erstes eingegeben. Zeile 570 bis 700 enthält Namen, Vornamen und Geburtsjahr eines jeden Mitglieds ihrer Trainingsgruppe.

Zeile 710 bis 970 erstellt eine Liste für den Übungsleiter. Das Pro-



▲ *Bild 1. Anhand der Startkarte wird jeder Teilnehmer einer Wettkampfgruppe zugeordnet*



TI 99

gut vorbereitet mit dem

Schwimmwettkampf gut vorbereitet mit dem TI99

gramm zeigt — aufgrund der vorher eingegebenen Daten — die jeweilige Wettkampfgruppe und nacheinander die Namen aller Gruppenmitglieder an. Cordula Berger muß lediglich eingeben, ob derjenige bei der angezeigten Schwimmart in der Wettkampfgruppe startet oder nicht. Im positiven Fall gibt sie noch die bisherige Bestzeit des Teilnehmers an.

Mit Zeile 980 bis 1460 druckt sie eine übersichtliche Liste aus, die bei jeder Wettkampfgruppe die Namen derer enthält, die in dieser Gruppe starten. Danach (Zeile 1470 bis 1880) werden die Mitteilungen für jeden Teilnehmer gedruckt (Bild 2). Eines bedauert Cordula Berger: Die Startkarten muß sie weiterhin per Hand ausfüllen, da sie leider wegen ihres Formats in keinen Drucker passen.

(Cordula Berger/kg)

TI 99/4A

```
100 DATA DIENSTAG,20 MAI 1984
110 DATA 18 UHR 15,DONNERSTAG,22
     MAI 1984,18 UHR 15
120 DATA SONNTAG,25 MAI 1984,14
    UHR
130 DATA 50,BRUST W,50,BRUST M,1
    00,FREISTIL W,100,FREISTIL M
140 DATA 50,RUECKEN W,50,RUECKEN
     M,100,BRUST W,100,BRUST M
150 DATA 50,SCHMETTERLING W,50,S
    CHMETTERLING M,100,RUECKEN W
    ,100,RUECKEN M
160 DATA 50,FREISTIL W,50,FREIST
    IL M,100,SCHMETTTERLING W,10
    0,SCHMETTERLING M
170 REM PROGRAMM FUER SCHWIMMVER
    ANSTALTUNGEN
180 REM **********************
190 REM *CORDULA BERGER     *
200 REM *BERLIN  45         *
210 REM **********************
220 CALL CLEAR
230 PRINT "PROGRAMM FUER SCHWIMM
    VERANSTALTUNGEN"
240 PRINT
250 INPUT "NAME DES WETTKAMPFS?
    ":W$
260 REM A=ANZAHL DER VERANSTALTU
    NGSTAGE
270 REM TAG$=WOCHENTAGE DER VERA
    NSTALTUNG
280 REM D$=DATUM
290 REM U$=UHRZEIT
300 DIM TAG$(3),D$(3),U$(3)
310 FOR I=1 TO 3
320 READ TAG$(I)
330 READ D$(I)
340 READ U$(I)
350 NEXT I
360 REM B=ANZAHL DER WETTKAEMPFE
370 REM S=STRECKE
380 REM LAGE$=SCHWIMMART
390 DIM S(20),LAGE$(20)
400 B=16
410 FOR I=1 TO B
420 READ S(I),LAGE$(I)
430 NEXT I
440 PRINT
450 REM M$=MAEDCHEN, J$=JUNGEN,
    JG=JAHRGANG
460 DIM M$(10),J$(10),JG(20)
470 FOR I=1 TO 8
480 READ M$(I)
490 READ JG(I)
500 NEXT I
510 FOR I=1 TO 6
520 READ J$(I)
530 READ JG(10+I)
550 NEXT I
560 GOTO 710
570 DATA SABINE MUELLER,1966
580 DATA YVONNE BAUM,1970
590 DATA ANNA GENT,1970
600 DATA SABINE SCHUTZ,1971
610 DATA SIMONE STULLE,1972
620 DATA MANUELA VOGEL,1973
630 DATA KATHRIN MEIER,1974
640 DATA BIRGIT MAAS,1974
650 DATA STEFAN SCHWAN,1971
660 DATA STEFAN WEIS,1971
670 DATA REINHARD SAND,1973
680 DATA WALTER ELER,1974
690 DATA CLAUS WIRT,1975
700 DATA ROBERT VOSS,1976
710 REM LISTE FUER DEN UEBUNGSLE
    ITER
720 DIM ST(20,20),BZ$(20,20)
730 PRINT "LISTE FUER UEBUNGSLEI
    TER"
740 FOR I=1 TO 15 STEP 2
750 PRINT S(I),LAGE$(I)
760 FOR J=1 TO 8
770 PRINT M$(J)
780 INPUT "J=JA,N=NEIN ":H$
790 IF H$="N" THEN 830
800 ST(J,I)=1
810 INPUT "BESTZEIT?":BZ$(J,I)
820 GOTO 840
830 ST(J,I)=0
840 NEXT J
850 NEXT I
860 FOR I=2 TO 16 STEP 2
870 PRINT S(I),LAGE$(I)
880 FOR J=1 TO 6
890 PRINT J$(J)
900 INPUT "JA=J,NEIN=N ":H$
910 IF H$="N" THEN 950
920 ST(J+10,I)=1
930 INPUT "BESTZEIT?":BZ$(J+10,I
    )
940 GOTO 960
950 ST(J+10,I)=0
960 NEXT J
970 NEXT I
```

Listing »Schwimmwettkampf«

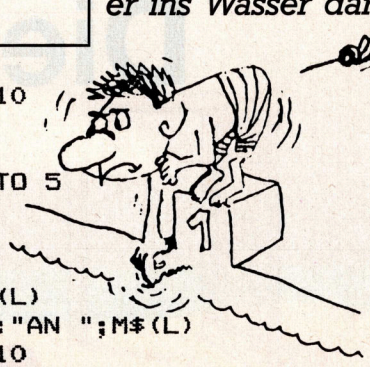
►

```
AN SABINE SCHUTZ

DIE WETTKAEMPFE NR.1 -4 FINDEM AM DIENSTAG 20 MAI 1984 18 UHR 15,
NR. 5-8 AMDONNERSTAG 22 MAI 1984 18 UHR 15,
NR.9-16 AM SONNTAG 25 MAI 1984 14 UHR STATT.
VERANSTALTUNGSORT FUER ALLE TAGE IST DIE GOETZSTRASSE.
WETTKAMPF NR.   1              50  METER BRUST W
WETTKAMPF NR.   5              50  METER RUECKEN W
WETTKAMPF NR.   15             100  METER SCHMETTTERLING W

 MIT FREUNDLICHEN GRUESSEN                    CORDULA BERGER
```

Bild 2. Jeder Teilnehmer erfährt genau, wann er ins Wasser darf



```
980 REM AUSDRUCK DER LISTE
990 OPEN #1:"PIO"
1000 PRINT "STARTLISTE FUER DAS
     ";W$
1010 PRINT #1:"STARLISTE FUER DA
     S ";W$
1020 PRINT
1030 PRINT #1
1040 PRINT TAG$(1),D$(1),U$(1)
1050 PRINT #1:TAG$(1),D$(1),U$(1
     )
1060 FOR I=1 TO 4
1070 PRINT #1:"WK ";I;" ";S(I);"
      METER ";LAGE$(I)
1080 GOSUB 1110
1090 NEXT I
1100 GOTO 1270
1110 K=I/2
1120 IF K=INT(I/2)THEN 1200
1130 PRINT S(I);" METER ";LAGE$(
     I)
1140 FOR J=1 TO 8
1150 IF ST(J,I)=0 THEN 1180
1160 PRINT M$(J),BZ$(J,I)
1170 PRINT #1:M$(J);TAB(20);BZ$(
     J,I)
1180 NEXT J
1190 GOTO 1260
1200 PRINT S(I);"METER ";LAGE$(I
     )
1210 FOR J=1 TO 6
1220 IF ST(10+J,I)=0 THEN 1250
1230 PRINT J$(J),BZ$(10+J,I)
1240 PRINT #1:J$(J);TAB(20);BZ$(
     10+J,I)
1250 NEXT J
1260 RETURN
1270 PRINT
1280 PRINT #1
1290 PRINT TAG$(2),D$(2),U$(2)
1300 PRINT #1:TAG$(2),D$(2),U$(2
     )
1310 PRINT
1320 PRINT #1
1330 FOR I=5 TO 8
1335 PRINT #1:"WK ";I;" ";S(I);"
      METER ";LAGE$(I)
1340 GOSUB 1110
1350 NEXT I
1360 PRINT
1370 PRINT #1
1380 PRINT TAG$(3),D$(3),U$(3)
1390 PRINT #1:TAG$(3),D$(3),U$(3
     )
1400 PRINT
1410 L=0
1420 PRINT #1
1430 FOR I=9 TO 16
1435 PRINT #1:"WK ";I;" ";S(I);"
      METER ";LAGE$(I)
1440 GOSUB 1110
1450 NEXT I
1460 PRINT
1470 FOR I=1 TO 5
1480 PRINT #1
1490 NEXT I
1500 L=L+1
1510 PRINT M$(L)
1520 PRINT #1:"AN ";M$(L)
1530 GOSUB 1810
1540 FOR I=1 TO 15 STEP 2
1550 IF ST(L,I)=0 THEN 1580
1560 PRINT S(I),LAGE$(I)
1570 PRINT #1:"WETTKAMPF NR. ";I
     ,S(I);" METER ";LAGE$(I)
1580 NEXT I
1590 GOSUB 1780
1600 IF L=8 THEN 1620
1610 GOTO 1470
1620 PRINT
1630 L=0
1640 FOR I=1 TO 5
1650 PRINT #1
1660 NEXT I
1670 L=L+1
1680 PRINT J$(L)
1690 PRINT #1:"AN ";J$(L)
1700 GOSUB 1810
1710 FOR I=2 TO 16 STEP 2
1720 IF ST(10+L,I)=0 THEN 1750
1730 PRINT S(I),LAGE$(I)
1740 PRINT #1:"WETTKAMPF NR. ";I
     ,S(I),LAGE$(I)
1750 NEXT I
1760 PRINT #1
1761 PRINT #1:"MIT FREUNDLICHEN
     GRUESSEN
     CORDULA BERGER"
1762 IF L<7 THEN 1640
1764 GOTO 1870
1780 PRINT #1
1790 PRINT #1:" MIT FREUNDLICHEN
      GRUESSEN                    C
     ORDULA BERGER"
1800 RETURN
1810 PRINT #1
1820 PRINT #1:"DIE WETTKAEMPFE N
     R.1 -4 FINDEM AM ";TAG$(1);
     " ";D$(1);" ";U$(1);","
1830 PRINT #1:"NR. 5-8 AM";TAG$(
     2);" ";D$(2);" ";U$(2);","
1840 PRINT #1:"NR.9-16 AM ";TAG$
     (3);" ";D$(3);" ";U$(3);" S
     TATT."
1850 PRINT #1:"VERANSTALTUNGSORT
      FUER ALLE TAGE IST DIE GOE
     TZSTRASSE."
1860 RETURN
1870 CLOSE #1
1880 STOP
```

TI Speech

Der Traum vom sprechenden Computer

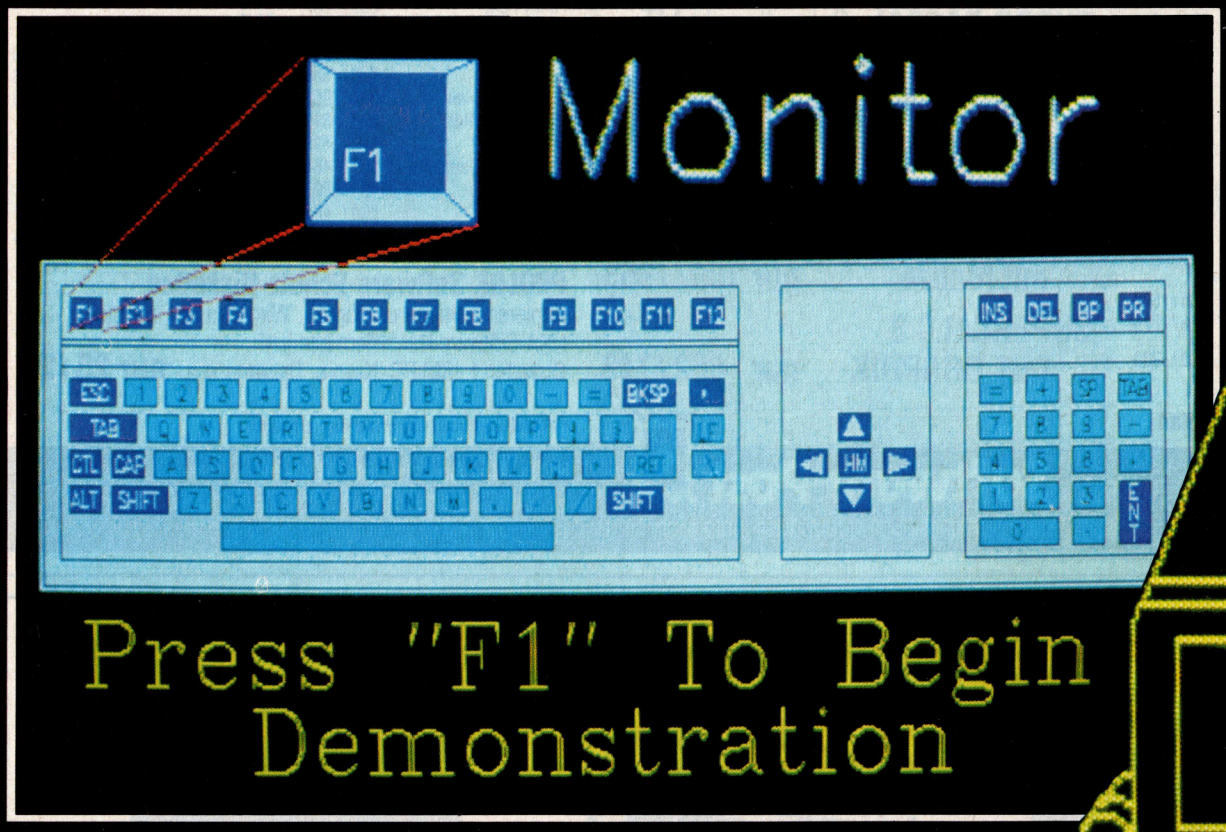


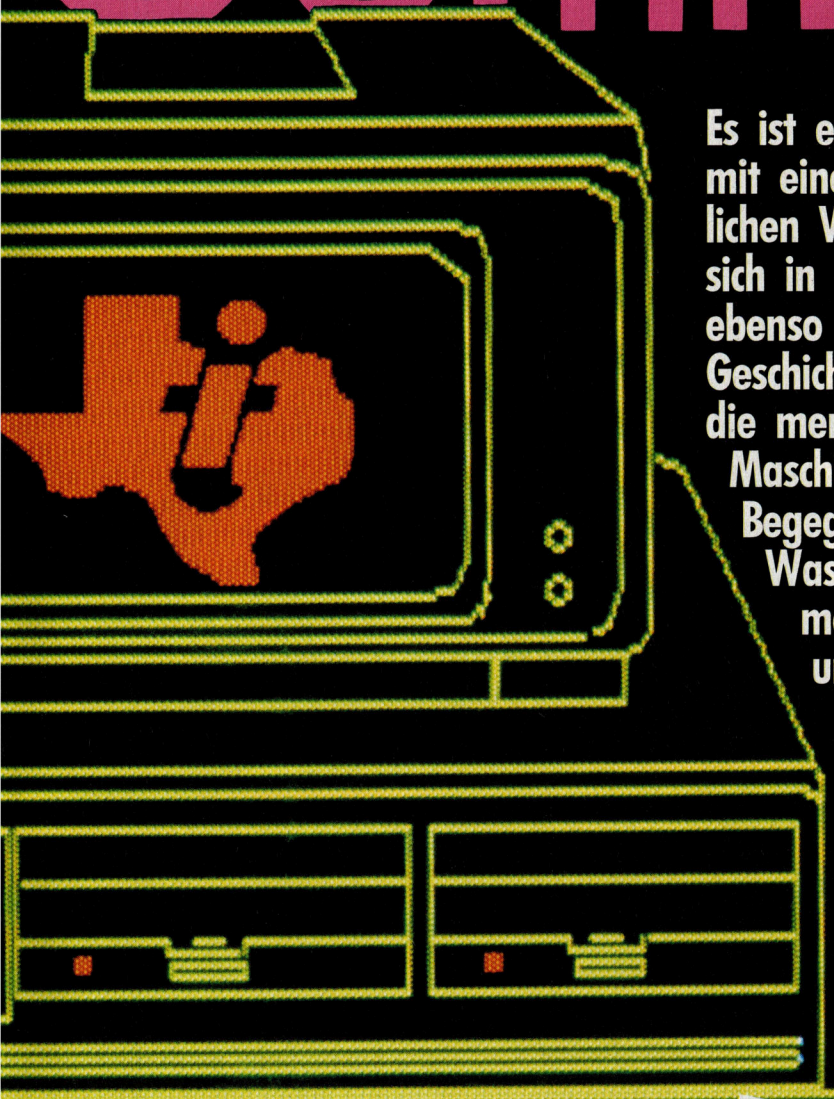
Bild 1. Aus der »Diaschau« des Demoprogramms: Hinweis auf die Funktionstaste zum Start der Sprachdemonstration

Die Einsatzgebiete sind vielfältig. Grundsätzlich geht es darum, dem Computer per Stimme Befehle zu geben, ohne dabei die Tastatur berühren zu müssen oder die Schreibweise von Befehlen zu wissen. Die »Speech Command Software« für den TI Professional und seinen portablen Bruder zeigt, was hier heute schon möglich ist: Der Computer erkennt einzelne gesprochene Wörter und reagiert darauf, er kann selbst sprechen oder besser gesagt, Sprachlaute erzeugen, und er kann die Stimme seines Benutzers wie ein Tonband aufzeichnen und wiedergeben und wird damit zum digitalen Diktiergerät oder Anrufbeantworter.

Den ersten Kontakt schließt man mittels eines Demonstrationsprogrammes, das sich beim Einschalten des Computers von selbst startet: Es beginnt eine eindrucksvolle »Farbdiaschau« (Bild 1), die — ganz nebenbei — die bestechende Farbgrafik des TI Professional vorführt. Zwischendurch erscheint immer wieder auf dem Bildschirm die Aufforderung die Taste <F1> zu drücken, um die Sprachdemonstration zu starten. Sobald man dieser Aufforderung nachkommt, beginnt der Computer in beinahe aufdringlich jovialem Tonfall und etwas quäkend, sich vorzustellen. Es handelt sich offensichtlich um die aufgenommene Stimme eines Verkaufsmanagers, die recht gut verständlich ist und sich ein wenig anhört, wie von einem Kassettenrecorder mit leicht klemmender Kassette wiedergegeben. Dafür sind die Algorithmen zur Kompression der Sprache verantwortlich, mit denen man den Bitstrom auf ein erträgliches Maß reduziert.

Es erscheint Bild 2, wozu der Computer erklärt, man solle darauf

Command



Es ist ein alter Traum der Menschheit, mit einem anderen, nichtmenschlichen Wesen zu sprechen. Das schlägt sich in Märchen von sprechenden Tieren ebenso nieder wie in den zahlreichen Geschichten von sprechenden und die menschliche Sprache verstehenden Maschinen und Robotern oder auch der Begegnung mit Außerirdischen.

Was immer Utopie blieb, scheint heute möglich: Computern beizubringen, unsere Sprache zu verstehen und in ihr zu antworten.

System Unit

Noch sind wir zwar von diesem Ziel weit entfernt, doch neuere Entwicklungen zeigen erstaunliche Möglichkeiten.

Bild 2. Eintrainieren von »Farbnamen« per Mikrofon

eyboard

achten, daß das Mikrofon eingeschaltet ist und möge die jeweils auf dem Bildschirm erscheinenden Farbnamen in dasselbe hineinsprechen.

Dieser Vorgang heißt »trainieren«, er dient dazu, den Computer die Lautmuster von Befehlswörtern speichern zu lassen. Spätestens hier wird deutlich, was dieses System eigentlich nur tut: Es speichert Klangmuster und versucht, sie wieder zu erkennen. Ob man die angeschriebenen englischen Farbnamen benutzt, statt »Indigo« lieber »Blau« sagt oder vielleicht »Kuh«, spielt keine Rolle. Der Computer »versteht« die Worte nicht, er kann sich bloß merken, wie sie klingen.

Hier taucht der Begriff »to enroll« auf. Er bedeutet »einschreiben«, was — wie wir später noch sehen werden — von »updating« zu unterscheiden ist.

Ich habe dem Computer die Worte »Blau«, »Rot«, »Purpur«, »Grün«, »Türkis«, »Gelb«, »Weiß«, »Stop« und »Go« gesagt. Jedesmal, wenn man ein Wort deutlich genug ins Mikrofon gesprochen hat, springt die Anzeige des Farbnamens auf das nächste Feld. Sobald man das letzte Befehlswort eingegeben hat, bemerkt der Computer zur Verblüffung des Publikums (auf englisch): »Das war doch ganz einfach, nicht wahr!«

Es folgen weitere Erläuterungen und es erscheint dasselbe Bild noch einmal, diesmal aber kann man die eintrainierten Wörter in beliebiger Reihenfolge sagen, und der Computer gibt zu verstehen, daß er sie erkannt hat, indem er den betreffenden (englischen) Farbnamen anschreibt. Diese Demonstration klappt meist fehlerlos. Erkennt der Computer ein Wort nicht gleich, so wiederholt die Versuchsperson es meist automatisch mit mehr Nachdruck, was in der Regel zum Erkennen führt. Falls es doch nicht klappen sollte, kann man die Trainingsphase wiederholen.

Gibt man zweimal den Befehl »Go« (was ich für »continue« gesagt habe), so erscheint ein neues Bild (3). Hier kann man nun durch Eingabe eines Farbnamens aus der dargebotenen Liste von Vorträgen gesprochene Erläuterungen über die Historie von TI, die Hardware des Systems und anderes mehr abrufen. Zunächst erscheint ein weiteres Menü, in dem man zu dem betreffenden Gebiet verschiedene Kapitel auswählen kann, beispielsweise über die Zentraleinheit oder Peripherie, wenn man etwas über die Hardware wissen will. Hat man wieder das entsprechende Wahlwort ins Mikrofon gesagt, so erscheint eine Folge von »Farbdias«, begleitet von dem entsprechenden gesprochenen Vortrag.

Dieser recht eindrucksvolle »Dialog« mit dem Computer zeigt das recht simple Grundprinzip der »Spracherkennung«: Man trifft mit dem Computer eine Vereinbarung darüber, welchem Befehl oder welcher Menüauswahl ein Wortklang zugeordnet werden soll. Daraufhin kann man — anstatt eine Auswahltaste zu drücken — den Computer durch gesprochene Befehle steuern. So simpel dies Prinzip auch ist — es zeigt eine neue Dimension des Umgangs mit dem Computer — als Multimediamaschine, wozu die exzellente Farbgrafik ebenso beiträgt wie die Steuerung eines gesprochenen Vortrages durch gesprochene Anweisungen (Bild 4). Bricht man nun das Demonstrationsprogramm ab, so bemerkt man erstaunt, daß es in Basic geschrieben ist.

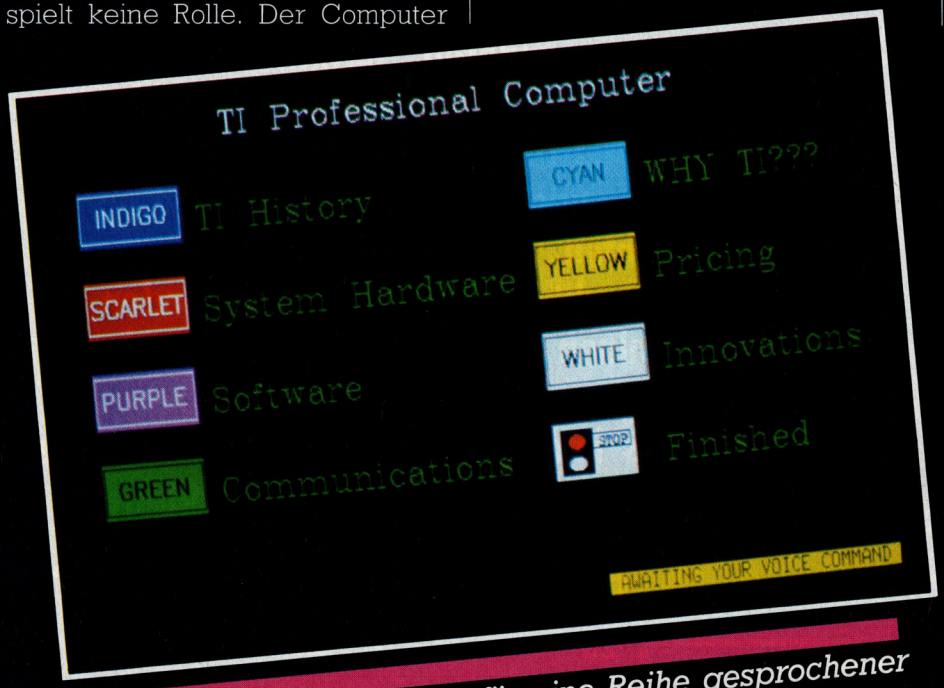


Bild 3. Menü für eine Reihe gesprochener Vorträge, aufrufbar per gesprochenem Farbnamen

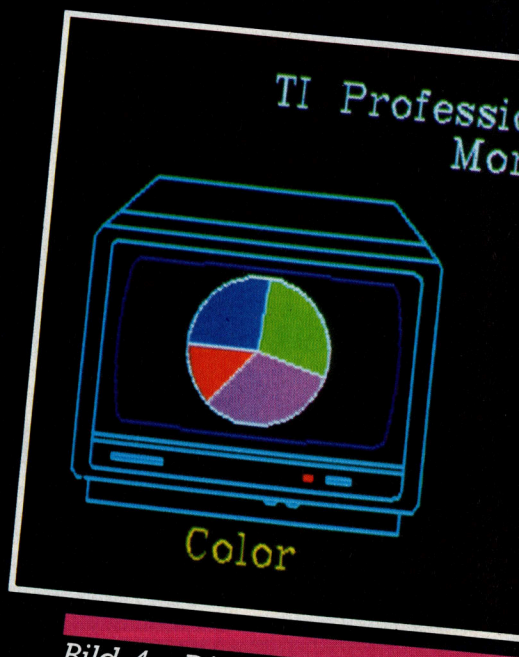


Bild 4. »Dia« zum Vortrag über Bildschirminterfaces: farbig od

Gedacht ist das zugrundeliegende System als eine Schicht im Betriebssystem, die es ermöglicht, jedes beliebige Programm, ohne dieses irgendwie verändern zu müssen, per Spracheingabe zu steuern. Dies wird dadurch erreicht, daß — softwaremäßig — die Spracherkennung zwischen Tastatur und Anwenderprogramm zwischengeschaltet wird. Es ist daher möglich, durch gesprochene Befehle einem laufenden Programm — wie etwa Lotus 1-2-3, Multiplan — oder auch dem Basicinterpreter — per Spracheingabe dieselben Befehle einzuspeisen, als hätte man diese über die Tastatur eingetippt. Dafür und für weitere Anwendungen gibt es ein Dienstprogramm, mit dem Wortschätze und ihnen zugeordnete Befehlseingaben eingerichtet werden können.

Die dazu erforderliche Ausstattung besteht aus zwei miteinander verbundenen Hardware-Platinen, einem Handbuch, einer Diagnosediskette und einer Kopfhörer-Mikrofonkombination oder einem

Handmikrofon und Wiedergabe über den eingebauten Lautsprecher. Das Softwarepaket besteht aus einem weiteren Handbuch und zwei Disketten. Uns lag eine vorläufige Fassung des Handbuchs als Fotokopie vor, die einen Umfang von zirka 150 Seiten haben dürfte und eine gut verständliche und sorgfältig gestaltete (englische) Anleitung bot.

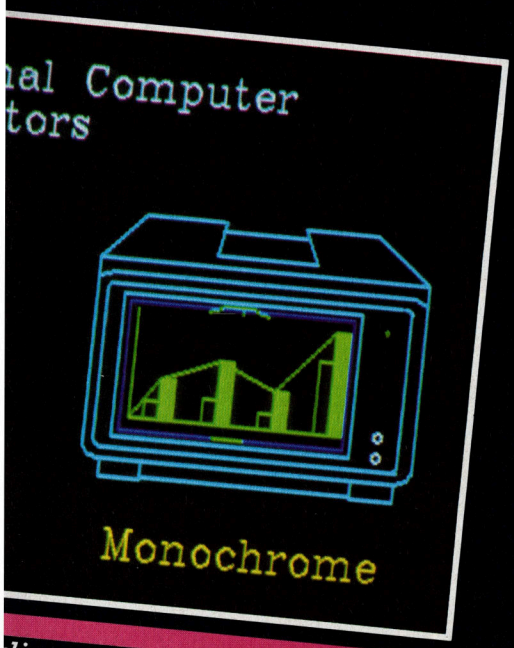
In den USA kann das System auch direkt zwischen die Telefonbuchse und das Telefon geschaltet werden, indem der Computer mit der Telefonbuchse verbunden und der Telefonapparat am Computer angesteckt wird. Damit wird der TI zu einer sehr leistungsfähigen intelligenten Kommunikationsstation. In der Bundesrepublik ist nur ein sehr rudimentärer Einsatz als »Anrufbeantworter« zulässig, man hat sich um entsprechende Lizenzen bemüht, darf aber nicht alle Einsatzmöglichkeiten nutzen. Stehen da nicht Behördenauflagen dem technischen Fortschritt im Wege?

Um nun die erwähnte »Zwischenschicht« im Betriebssystem einzurichten, gibt man einfach den Befehl »PCSPEECH«; dadurch wird eine Batch-Datei aufgerufen, die das Installieren eines interruptgesteuerten Treibers besorgt.

Das Dienstprogramm, das für den Gebrauch beziehungsweise die Vorbereitung von Sprachverarbeitung erforderlich ist, wird durch »SCS« aufgerufen, woraufhin das Menü in Bild 5 erscheint.

Zunächst wähle ich den Punkt »Transparent Keyboard Vokabulary Utility«. Es erscheint ein neues Schirmbild, in dessen Menü man zunächst eine Vokabulardatei anlegen oder aufrufen kann. Hat man früher bereits eine solche Datei angelegt, so erscheint als Vorschlag der Name der zuletzt bearbeiteten Datei, und man braucht lediglich <RETURN> zu drücken, um sie zu laden (auch, wenn der Computer zwischenzeitlich abgeschaltet war). Das zeigt, daß auch diese Software ein hohes Maß an benutzerorientiertem Komfort bietet.

Mit diesem Programm kann man bis zu neun Vokabulardateien zu je 50 Wörtern anlegen — also bis zu 450 gesprochene Befehle erfassen. Da zwischen den einzelnen Wortschätzen eine Umschaltmöglichkeit durch »Schaltwörter« besteht, kann im Endeffekt ein gesprochener Befehlsumfang von 450 Begriffen genutzt werden. Mit der »transparenten Tastatur« ist die schon erwähnte Erscheinung gemeint, daß die gesprochene Eingabe eines Befehls dieselbe Wirkung hat, als hätte man auf der Tastatur die dazu vereinbarte Zeichenfolge (auch eine ganze Kette von Befehlen) eingegeben. Das entsprechende Anwendungsprogramm »glaubt« daher, die entsprechenden Anweisungen von der Tastatur empfangen zu haben — deshalb brauchen normale Standardprogramme in keiner Weise verändert werden, um von dieser Möglichkeit Gebrauch machen zu können. Diese Zeichen werden bei der Aufstellung eines Vokabulars im Dienstprogramm in der zweiten Spalte für jeden Begriff eingegeben. Steuerzeichen werden als »Controlcodes« etwa in der Form »^0D« (hexadezimal 13) für Carriage-Return (Return-Taste) eingegeben.



die erhältlichen
r monochrom

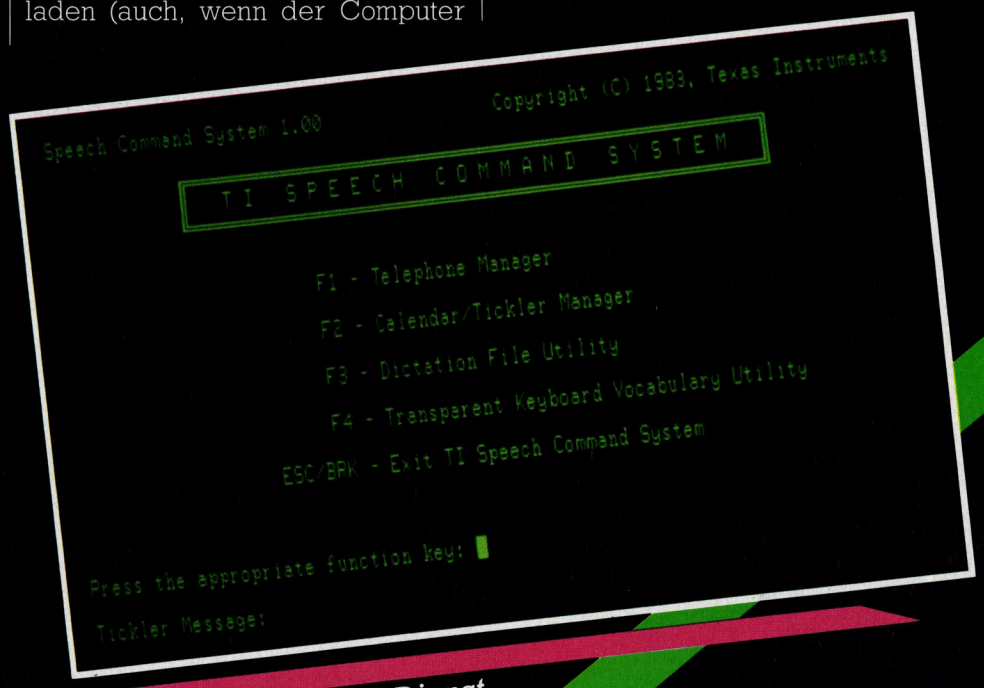


Bild 5. Menü des Dienstprogramms für Sprachverarbeitung

Jedes Vokabular wird durch zwei Dateien dargestellt, die jeweils den gleichen Namen, aber verschiedenartige Namenserweiterungen haben. Die erste Datei enthält schriftliche Bezeichnungen der akustischen Befehle (die der Identifikation beim Trainieren des Wortschatzes dienen) sowie diejenigen Zeichen, die (scheinbar) von der Tastatur ausgesandt werden sollen, wenn der betreffende Befehl erkannt wurde. Die Namenserweiterung lautet hier ».HDR«, also etwa »BASIC.HDR«.

Die zweite Datei, an deren Namen ».VOC« angehängt wird, enthält die Aufzeichnungen der gesprochenen Befehle in einem numerischen Code. Diese »Voiceprint« (Stimmabdruck) genannten Aufzeichnungen benutzt der Computer zum Vergleich mit den jeweils über Mikrofon empfangenen Sprachsignalen, um den jeweiligen Befehl zu erkennen. Dazu muß aber für jede gesprochene Anweisung zunächst einmal eine Aufzeichnung des Wortklanges erfolgen — und zwar von derjenigen Person, die den Computer be-

nutzen will. Leider ist das System nicht — oder höchstens einmal zufällig — in der Lage, gesprochene Befehle zu erkennen, wenn diese von einer anderen Person eingegeben werden als derjenigen, die den Computer »trainiert« hat. Vergleichen Sie hierzu den Artikel »Spracherkennung und ihre Grenzen«.

Zum »Trainieren« des Systems wird das Dienstprogramm benutzt. Zunächst gibt man die gewünschten Befehle und die durch sie auszulösenden Codes ein (Bild 6).

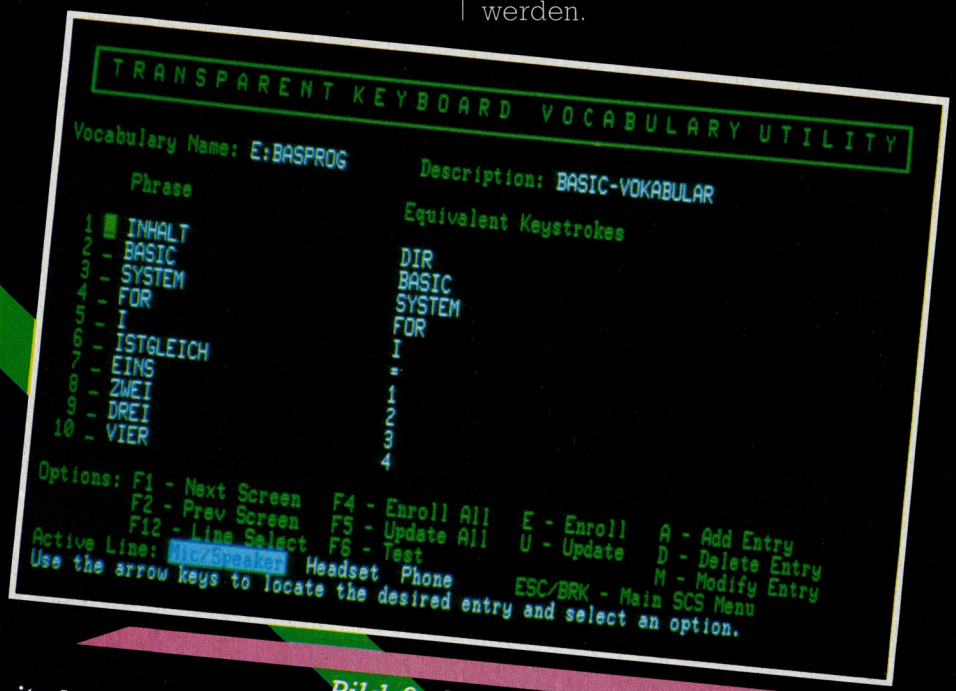


Bild 6. Auszug des Textvokabulars, mit dem verbal ein Basicprogramm editiert werden kann

Anschließend erfolgt das »Eintrainieren« der gesprochenen Begriffe. Danach schließt sich eine »Updating«-Phase an, bei der sich der Computer jedes Wort mehrmals vorsprechen läßt. Der Zweck: Man spricht die Worte nicht immer gleich aus, Tonfall, Geschwindigkeit, Betonung können sich ändern — ein gesprochenes Wort ist eine »unscharfe« Größe. Durch mehrfaches Sprechen bildet der Computer einen Mittelwert, um das Wort sicherer erkennen zu können. Ich habe beobachtet, daß dies tatsächlich eine Verbesserung der Worterkennung bewirkt.

Danach kann man mit der Betriebsart »Test« die Worterkennung überprüfen. Das Programm gibt dabei durch eine Farbskala und einen numerischen Wert von 1 bis 10 bekannt, wie gut oder sicher ein Begriff erkannt wurde. Rot bedeutet: »Sehr schlecht«, Grün dagegen: »Sehr gut«, Gelb: »Mittelmäßig« (Bild 7). Am besten erkannte der Computer ironischerweise ein nichtverbales Geräusch: das Klopfen gegen das Mikrofon. Das ist insofern verständlich, als dieses mechanisch erzeugte Geräusch immer genau gleich klingt, während gesprochene Laute durch die oben genannten Faktoren, unterschiedlichem Abstand zwischen Mund und Mikrofon, verschiedenartige Akustik des Raumes, Hintergrundgeräusche und so weiter verändert werden.

Bild 7. Prüfphase, in d die Sicherheit der Be

Probleme hatte ich, als ich englische und deutsche Begriffe durcheinander eingab. Relativ viele davon wurden in der anschließenden Testphase nicht oder nur sehr schlecht erkannt. Durch einiges Probieren merkte ich, daß ich offenbar unbewußt manche deutschen Wörter mit englischem und manche englischen mit deutschem Akzent eingegeben hatte. Ein in deutsch mit geschlossenem »E« wie in »Esel« gesprochenes »next« wurde schlecht oder nicht erkannt, wenn ich es mit breitem »äh« eingab und umgekehrt.

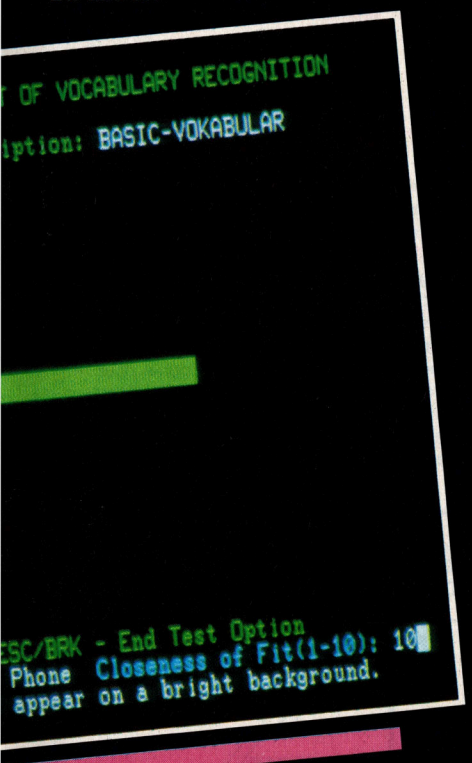
Auch kommen durchaus Verwechslungen ähnlich klingender Worte wie »next« und »sechs« vor, umgekehrt wurde »hallo«, wenn als deutsches »Hallo« eingegeben, manchmal auch erkannt, wenn ein amerikanisches »hello« ertönte.

Das Handbuch diskutiert diese Problematik und empfiehlt, Wörter zu wählen, die sich klanglich deutlich unterscheiden, beispielsweise soll man nicht »right« und »write« gleichzeitig benutzen. Außerdem würden Wörter um so sicherer erkannt, je länger sie seien, weil längere Wörter mehr Information (mehr kennzeichnende Silben) enthalten. Auch soll man keine Wörter benutzen, die Teile von anderen Wörtern sind, wie etwa »Haus« und »Hausbau«. Zulässig sind dagegen Begriffe wie »Haustür« und »Hausbau«, weil keiner der beiden als Teil des anderen vorkommt.

Erstaunlich ist auf der anderen Seite, daß das System die trainierten Worte in der Regel auch dann erkennt, wenn sie in einem schnell gesprochenen Satz auftauchen. So wird in »bitte den Inhalt« ohne weiteres »Inhalt« erkannt, oder in »Ich will Basic programmieren« das Wort »Basic«. Das Handbuch empfiehlt, zur Vorbereitung darauf beim »Updating« der Wortklänge die Wörter in entsprechende Sätze eingebettet zu wiederholen, damit sich das System die dabei entstehenden leichten Klangänderungen merken kann. Mir scheint allerdings, daß dies im Amerikanischen eine größere Rolle spielt als im Deutschen.

Wenn man Wörter, von denen man merkt, daß der Computer sie leicht verwechselt, vermeidet, kann man eine recht zuverlässige

Spracheingabe realisieren, da das System bei nicht erkannten Wörtern nicht aufs Geratewohl das nächstähnliche »erkennt«, sondern klar mit der Meldung antwortet: »Nicht erkannt«. Um ein Vokabular zu benutzen, muß man mit »TPK-SETUP« wiederum einen entsprechenden Treiber installieren und dann das Programm »TPK« (Trans-Parent Keyboard) aufrufen, um die entsprechenden Vokabulardateien zu laden.



r
hlserkennung angezeigt wird

Nach diesen Vorbereitungen und nachdem ich ein entsprechendes Vokabular eingegeben hatte, gelang mir etwas, das mir — warum auch immer — als noch viel phantastischer erschien als alles bisherige:

Ohne die Tastatur des Computers auch nur ein einziges Mal zu berühren, »rief« ich (im wahrsten Sinne des Wortes) den Basic-Interpreter vom Betriebssystem aus auf, gab — per Sprache — ein Programm ein und ließ es laufen (Bild 8). Es gab zwar ein paar Mal Fehler, indem auf dem Bildschirm statt des Befehls »NEXT« die Ziffer »6« erschien, aber mit Hilfe des Befehls »BACK« (Backspace) ließ sich dies per Sprache wieder löschen und schließlich gelang es mir, »NEXT« auf dem Bildschirm zu bringen. Anschließend konnte ich das Programm starten, indem ich »RUN RETURN« ins Mikrofon rief.

Damit könnte beispielweise ein Behinderter der keine Tastatur bedienen kann, einen Computer programmieren — oder jemand, der an seinem Arbeitsplatz keine Hände frei hat, einen Computer bedienen.

Sicher ist ein Umfang von 450 Wörtern zu gering, um die Tastatur gänzlich zu ersetzen, auch ist das Trainieren von so vielen Befehlen etwas mühsam. Auch ist die Eingabesicherheit deutlich geringer — aber es geht!

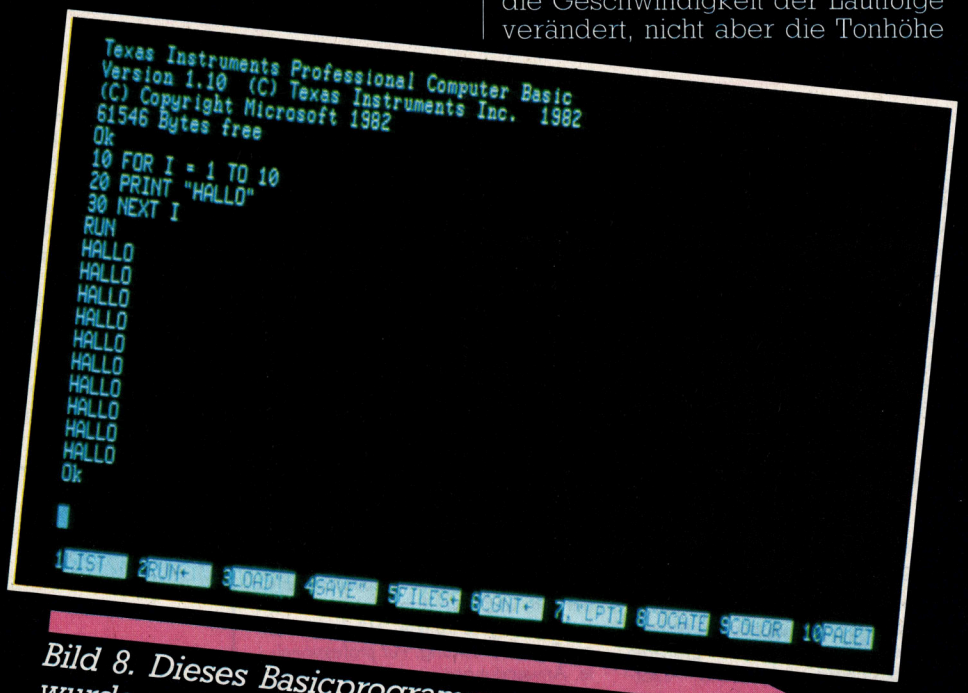


Bild 8. Dieses Basicprogramm wurde verbal eingegeben, ohne die Tastatur zu berühren

Angesichts des zu Anfang besprochenen Demoprogramms stelte ich dem Distributor die Frage, ob es denn für den Benutzer auch möglich wäre, seine eigenen sprechenden Programme zu erstellen. Antwort: Noch nicht, hierzu befindet sich aber ein »Speech development kit« in Vorbereitung — sobald dieses vorhanden ist, kann man dann — ähnlich wie es jetzt schon bei der Betriebsart »Diktiergerät« möglich ist — vom Computer aufgenommene Texte von eigenen Programmen gesteuert wiedergeben lassen. Erst damit würde der TI zum wirklich sprechenden Computer, mit dem man — bei entsprechend geschickter Programmierung — sicherlich sogar schon einfache Frage-Antwort-Dialoge aufbauen könnte.

Mit der Betriebsart »Dictation File Utility« kann man den Computer als Diktafon benutzen. Man spricht einfach ins Mikrofon, als ob man ein Tonband bespricht. Anschließend kann die Sprachaufzeichnung wieder abgehört werden. Sie ist ziemlich gut zu verstehen, allerdings klingt die Stimme etwas krächzend und merkwürdig »elektronisch verzerrt«. Für den Zweck »Diktiergerät« reicht die Qualität aus, ist aber schlechter als von einem Bandgerät. Interessant: Zu rasante Sprecher können bei der Wiedergabe gebremst, Langweiler beschleunigt werden — ohne daß der vom Tonband bekannte »Mickymouse«-Effekt eintritt, denn es wird zwar die Geschwindigkeit der Lautfolge verändert, nicht aber die Tonhöhe der Stimmlage. Allerdings nimmt dadurch die Verständlichkeit der Sprache etwas ab. Eine weitere Besonderheit: Sprechpausen werden, falls der Umgebungslärmpegel nicht zu hoch ist, nicht mit aufgenommen — sie entfallen bei der Wiedergabe. Man braucht also während des Diktats bei längeren »Denkpausen« das Computer-Diktiergerät nicht einmal abzuschalten.

Der Computer als Diktiergerät: Krächzend, aber verständlich

Sehr schlecht zu verstehen sind Stimmen, die von einem Lautsprecher aufgenommen wurden. Ich habe probehalber ein paar Minuten Werbefunk aufgenommen, in dem verschiedene männliche und

Fortsetzung auf Seite 205

Fortsetzung von Seite 201

weibliche Sprecher vorkommen, indem ich im Zimmer ein laut eingestelltes Radio laufen ließ. Im Gegensatz zu den direkt ins Mikrofon gesprochenen Texten waren die Stimmen kaum mehr zu verstehen. Zwischendurch vorkommende Musikabschnitte wurden zu rumpelnden Geräuschen verstümmelt. Dies kommt daher, daß die Sprachaufzeichnung besondere Algorithmen zur Kompression verwendet, die darauf spezialisiert sind, die kennzeichnenden Frequenzen und andere Eigenschaften der gesprochenen Laute herauszufiltern, um daraus später wieder als Sprache erkennbare Töne zu bilden. Singen oder Musizieren kann der Computer nicht! Es handelt sich also keineswegs um eine einfache Digitalisierung der aufgenommenen Geräusche, sondern um eine für Sprachwiedergabe bestimmte spezielle Codierung. Immerhin ist bei direkt ins Mikrofon gesungenen Kinderliedern die Melodie noch erkennbar, während eine kleine Pfeifeinlage sehr merkwürdig zurücktönt.

Während der Wiedergabe kann man durch Druck auf die entsprechenden Funktionstasten die Wiedergabe leiser und lauter, schneller oder langsamer werden lassen oder ein Stück zurück oder vorwärts springen, ganz zurück zum Anfang oder die Wiedergabe abbrechen.

Auf einer 320-KByte-Diskette können bis zu 16 Minuten Text, auf einer 10-Megabyte-Winchester bis zu acht Stunden Sprache aufgezeichnet werden.

Die Betriebsart »Telefon Manager« konnte ich nicht erproben, doch wird sich, nach den Worten eines Firmensprechers, dieses System in der Art eines herkömmlichen Telefon-Anrufbeantworters einsetzen lassen. Zwar könnte der Computer viel mehr — aber die Post läßt's nicht zu.

Zu den technischen Einzelheiten gab es noch keine Unterlagen, lediglich mündlich wurde uns mitgeteilt, daß für die Sprachverarbeitung der 32-Bit-Prozessor TMS 3200 verwendet wird, der mit 20 MHz Taktfrequenz läuft und die Sprache mit einer Rate von 8 kHz abtastet. Die erfaßten Worte werden dann in Echtzeit einer Fourier-Analyse unterworfen. Zur Codierung wird ein Verfahren namens LPC (Linear predictive coding) benutzt. Typische Einsatzgebiete wären beispielsweise industrielle Steuerungen (Paket-Verteilungsanlage) mit Spracheingabe.

Wenn das Paket, mit dem man eigene Programme sprechen lassen kann, noch hinzukommt, scheint mir dieses System eine hochinteressante computerisierte Multimedia-Einrichtung zu sein. Es ist geeignet für zahlreiche Arten der Unterweisung oder des programmierten Unterrichts, beispielsweise könnte man Programme erstellen, die ein Thema sowohl mit detaillierten Farbgrafiken als auch gesprochenen Texten erläutern. Gleichzeitig könnten an den Benutzer Fragen gestellt und seine Antworten ausgewertet werden. Denkbar wäre beispielsweise ein Einsatz im Geografieunterricht, auf dem Bildschirm könnte eine farbige Karte erscheinen, die per gesprochenem Text erläutert wird. Anschließend werden Fragen gestellt, die der Schüler durch Druck auf Funktionstasten oder durch Eingabe der Lösung beantwortet — je nach der Antwort kann die Erläuterung dann fortfahren oder das betreffende Thema wiederholen. (le)

Mehr wissen als andere

Als erfolgreicher Fachverlag für Elektronik- und Computerpublikationen arbeiten wir in einem zukunftssicheren Bereich und gehören zu den am schnellsten wachsenden Verlagen in Deutschland.

Zur Verstärkung unserer Redaktion suchen wir

Ingenieur/Informatiker/Programmierer für den Bereich Personal Computer und Software

Was Sie mitbringen müssen: Erfahrungen im Umgang mit Personal Computern, der dazugehörigen Peripherie, den wichtigsten Betriebssystemen, gute Kenntnisse über Standardsoftware für Personal Computer wie beispielsweise Datenbank-, Tabellenkalkulations-, Kommunikations- und Textverarbeitungsprogrammen.

Zu Ihrem Aufgabenbereich gehört das Testen von Personal Computern, der dazugehörigen Peripherie und Software, das Schreiben von verständlichen Artikeln über bestimmte Produkte und Produktbereiche sowie das »Aufspüren« und Recherchieren wichtiger Neuigkeiten in der Branche — alles in allem eine interessante Stelle, die für einen computererfahrenen Hochschulabgänger ebenso geeignet ist, wie für einen Praktiker.

Wenn Sie in einem jungen, erfolgreichen Team mitarbeiten wollen und an der ausgeschriebenen Position interessiert sind, sollten Sie uns kurzfristig Ihre Bewerbungsunterlagen zuschicken. Die ausgeschriebene Position ist entsprechend ihrer Anforderung gut dotiert. Engagierten Mitarbeitern bieten wir nicht nur eine interessante Tätigkeit, sondern auch attraktive Aufstiegschancen.

Telefonisch können Sie sich vorab unter der Rufnummer 089/4613-222, Michael Scharfenberger, informieren.

computer persönlich

Markt & Technik Verlag Aktiengesellschaft, Hans-Pinsel-Str. 2, 8013 Haar bei München

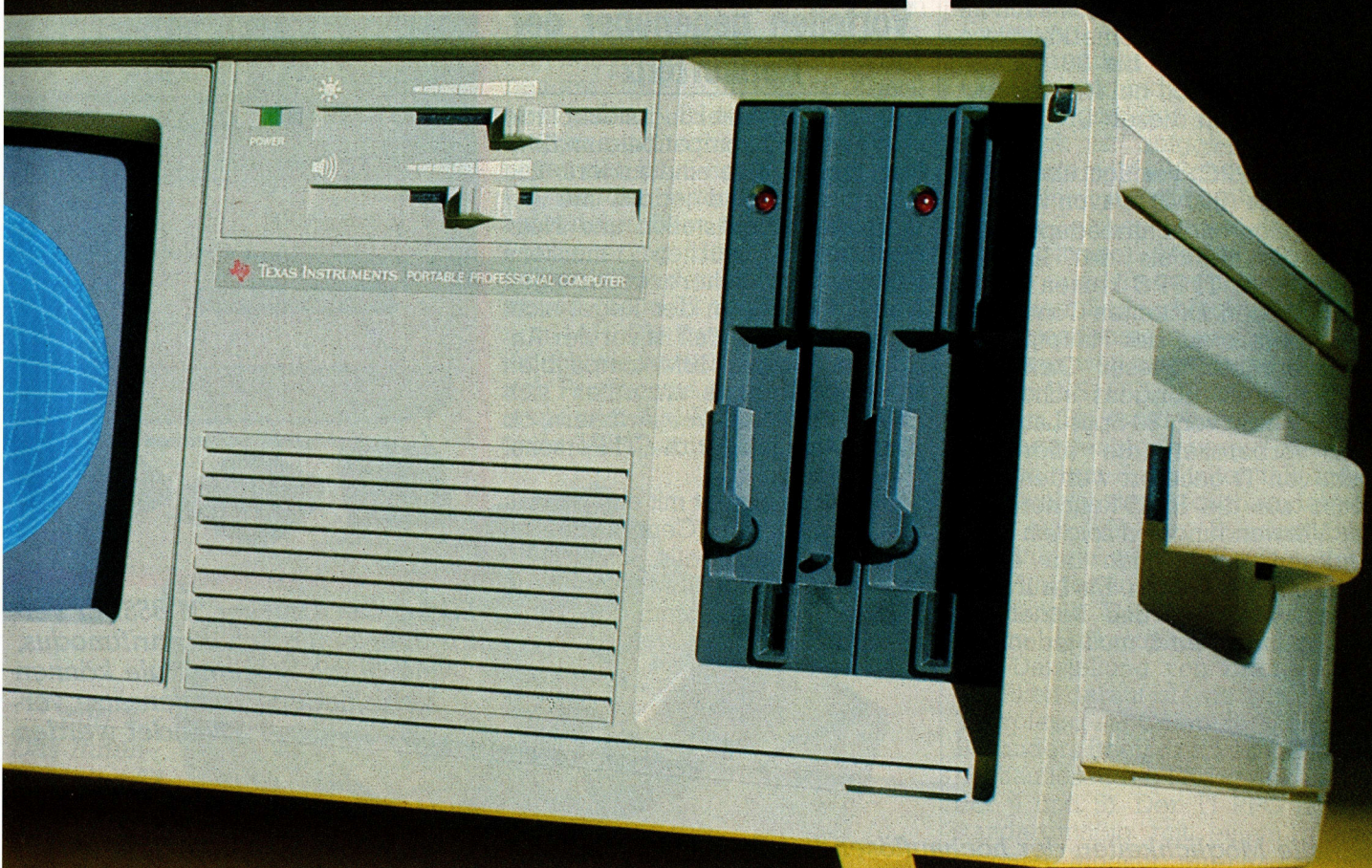
Der Portable Profess von TI

Technisch gesehen ist der TI Portable Professional Computer identisch mit dem schon bekannten TI Professional Computer. An dem waren im Test die gute Tastatur und die Super-Farbgrafik aufgefallen. Das ist hier nicht anders: Besonders sticht wieder das hervorragend scharfe und brillante Farbbild hervor. Statt »portabel« hätte es zwar besser »transportabel« heißen sollen, aber im großen und ganzen scheint er doch ein recht angenehmer Zeitgenosse zu sein.

Er sollte »transportabel« heißen, weil er mit seinen Abmessungen von 47 x 44 x 20 cm und einem Gewicht von etwa 15 kg nicht gerade sehr bequem zu tragen ist (Bild 1). Vielleicht wurden deshalb auch wohlweislich zwei Griffe angebracht. Dies ist jedoch einer der ganz wenigen Kritikpunkte am TI Portable Professional Computer, in allen anderen Punkten macht er einen zuverlässigen und soliden Eindruck.

Der Portable Professional Com-

sional Computer



puter (PPC) ist bitkompatibel zum TI Professional Computer. Das heißt, bis auf das Gehäuse und einen etwas anderen Aufbau sind die beiden Computer bis ins kleinste Teil identisch.

Hauptkonkurrent des TI PPC (Portable Professional Computer) ist — wie das fast bei jedem 8088-Rechner ist — der IBM Personal Computer, von dem vor kurzem in den USA auch eine tragbare Version angekündigt wurde (mit diesem hat er auch seine nicht besonders handlichen Abmessungen gemein). Im Vergleich zum IBM-PC hat er jedoch außer einem etwas günstigeren Preis auch eine etwas höhere Leistung zu bieten. Anfang März senkte IBM die Preise ganz erheblich; es wurden daraufhin bei TI Überlegungen angestellt, die Preise ebenfalls zu senken, um den Preisvorteil von 5 bis 8 Prozent weiterhin zu wahren. Die genauen Preise lagen bei Testabschluß jedoch noch nicht vor.

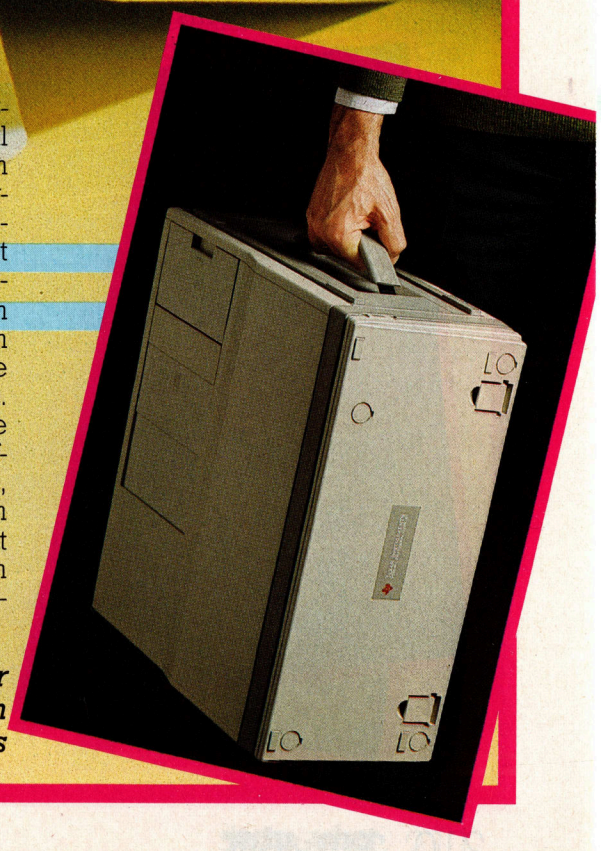


Bild 1. So sieht der PPC im zusammengebauten (transportablen) Zustand aus

Der 8088-Mikroprozessor im TI Portable Professional arbeitet mit einer Taktfrequenz von 5 MHz und kann durch den numerischen Coprozessor 8087 erweitert werden. Eine Fassung dafür ist bereits vorhanden. Mit seinen 5 MHz ist der TI etwas schneller als der IBM-PC (4,77 MHz). Das macht sich auch im normalen Betrieb bemerkbar, wo der PPC etwa 10 bis 20 Prozent schneller ist. Auch bei Diskettenzugriffen ist er in dieser Größenordnung schneller.

Der TI PPC verfügt über fünf Steckplätze (Bild 2). Davon wird einer für den Bildschirmadapter benötigt. Anders als der IBM-PC benötigt der TI PPC keinen Steckplatz für den Disketten-Controller. Er ist bereits auf der Hauptplatine untergebracht. Dieser Disketten-Controller ist übrigens schon für den Einsatz von 80-Spur-Laufwerken vorbereitet (die allerdings nicht von TI geliefert werden), genauso wie die BIOS-Routinen. Bei der Verwendung von solchen Laufwerken braucht man nur einen Jumper umzustecken. Ein Laufwerk (jenes, von dem das Betriebssystem geladen wird) muß jedoch eines mit 40 Spuren sein. Die normalen Laufwerke speichern 320 KByte unter DOS 1.25 und 360 KByte unter DOS 2.11. Die Diskettenlaufwerke sind im übrigen wesentlich ruhiger als beim TI PC.

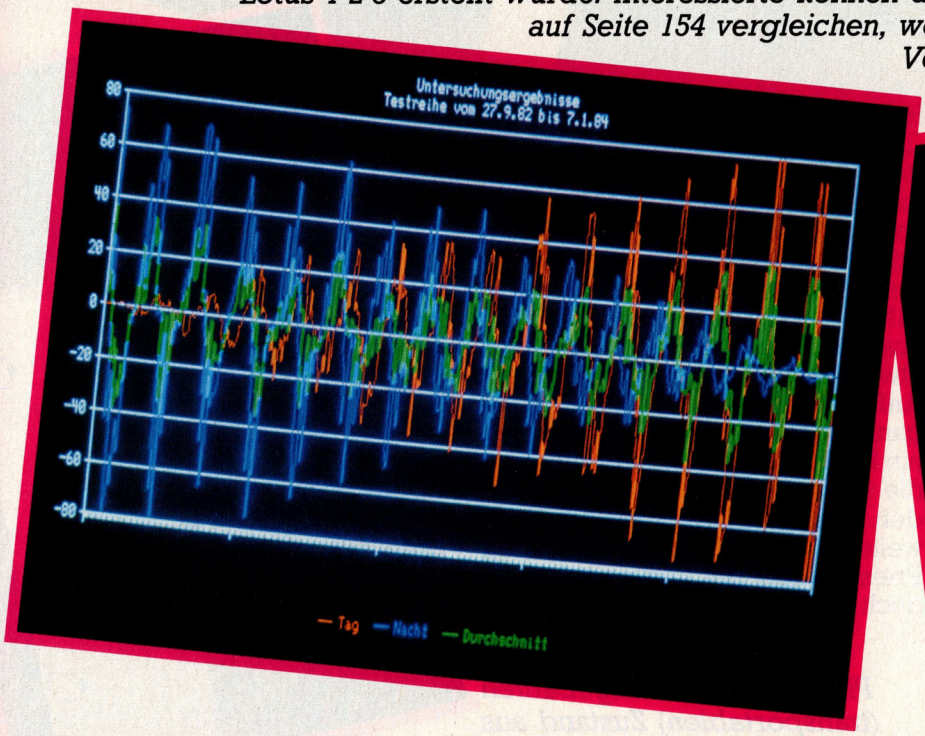
Hardware mit kleinen Vorteilen gegenüber der Konkurrenz

Die Erweiterungssteckplätze sind nicht voll kompatibel zum IBM-PC, drei Signale sind anders. Die Änderungen sind jedoch nur geringfügig. Schnittstellen und Netzwerkkarten dürften meistens passen; Speichererweiterungen werden jedoch in der Regel nicht laufen. In jedem Fall ist vor der Anschaffung einer IBM-kompatiblen Erweiterungskarte anzuraten, sich beim Anbieter zu vergewissern, ob die Karte auch für die TI-Personal Computer paßt.

Der Computer ist mit einer parallelen Schnittstelle (Centronics) ausgerüstet. Eine serielle Schnittstelle ist nicht serienmäßig eingebaut, sondern kann in einem der Steckplätze nachgerüstet werden.

Bild 8. Der Drucker TI 855. Er verfügt über einen Schönschriftmodus, und durch Steckmodule können verschiedene Zeichensätze verwendet werden

Bilder 4 und 5. Lotus 1-2-3 nutzt alle Möglichkeiten der hochauflösenden Farbgrafik. Bild 5 zeigt ein Balkendiagramm, das mit Lotus 1-2-3 erstellt wurde. Interessierte können das mit dem Bild in Ausgabe 5/84 auf Seite 154 vergleichen, wo das gleiche Bild in der IBM-PC-Version zu sehen ist.



In der Standardausführung ist der Portable Professional mit 128 KByte RAM ausgestattet. Der Speicher ist ausbaubar bis 768 KByte. Darüber hinaus ist der Rechner mit 8 KByte ROM ausgestattet. Dieses ROM enthält das BIOS (Basic I/O System), »device service routines« genannt. Diese Routinen erledigen die Ein- und Ausgabe des Systems, zum Beispiel das Lesen und Schreiben auf Diskette und Platte, Tastatureingabe, Bildschirmsteuerung etc. Sie entsprechen in ihren Funktionen den BIOS-Routinen des IBM-PC, werden jedoch etwas anders aufgerufen. Beim IBM-PC wird die Positionierung des Cursors zum Beispiel über den Interrupt 10H gesteuert; beim TI ist das der Interrupt 4AH. Die Registerbelegung ist jedoch wieder sehr ähnlich, so daß sich Programme in der Regel sehr leicht vom IBM-PC auf den TI-PPC (oder TI-PC) umstellen lassen.

Es ist eine interessante Erweiterungsplatine erhältlich, die Sprach-ein-/-ausgabe und -erkennung möglich macht (näheres darüber ist in einem separaten Artikel in diesem Heft zu finden).

Die Tastatur fällt beim TI Portable Professional Computer als erstes positiv auf (Bild 3). Sie hat eine gute Einteilung (Schreibmaschinenblock, darüber 12 Funktionstasten, Cursorblock und separate Zehnertastatur), und der Schreibmaschinenblock ist aufgebaut wie bei einer normalen Schreibmaschine (also ohne störende Größer- und Kleiner-Zeichen neben der linken Umschalttaste). Markierungen auf den F- und J-Tasten und der Taste 5 auf dem Zehnerblock erleichtern die Orientierung ohne Sichtkontrolle. Etwas weit abseits sind die vier Tasten INS, DEL, PAUS und PRNT plaziert. Sie würden mir über dem Cursortastenblock besser gefallen.

Nach dem Einschalten fällt sofort die Qualität der Bildschirmdarstellung auf. Mit der Standard-Bildschirmkarte werden Zeichen mit 7 x 9 Punkten in einer 9 x 12-Matrix dargestellt. Das Testgerät war mit einem Farbmonitor ausgestattet. Die Darstellung von Zeichen und Grafiken darauf ist (wie schon beim TI Professional Computer) sehr gut. Besonders beeindruckend sind die hochauflösenden Farbgrafiken (Bilder 4 und 5). Der TI PPC kommt standardmäßig mit einem monochromen Monitor, ein Farbmonitor ist optimal erhältlich. Beide Monitore können die hochauflösende Grafik mit 720 x 320 Punkten darstellen.

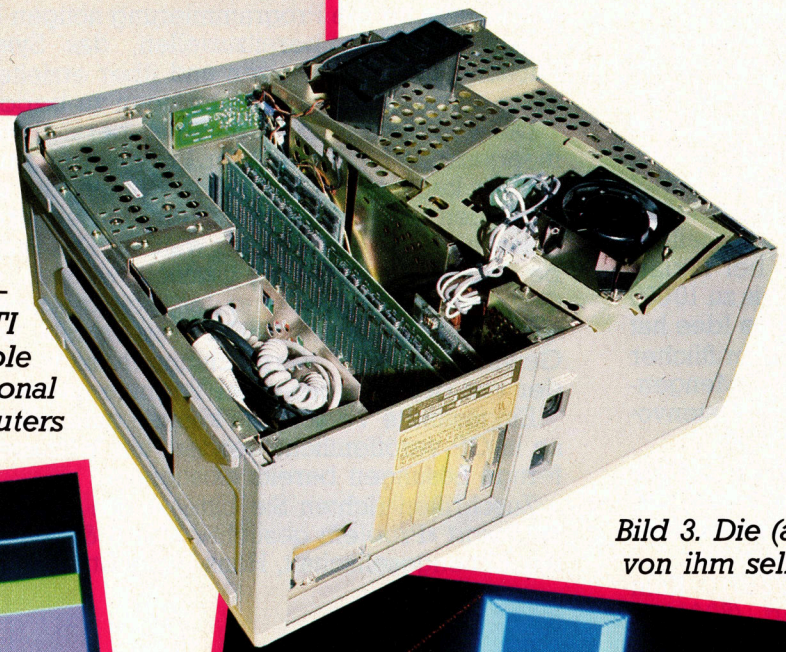


Bild 2. Das Innenleben des TI Portable Personal Computers

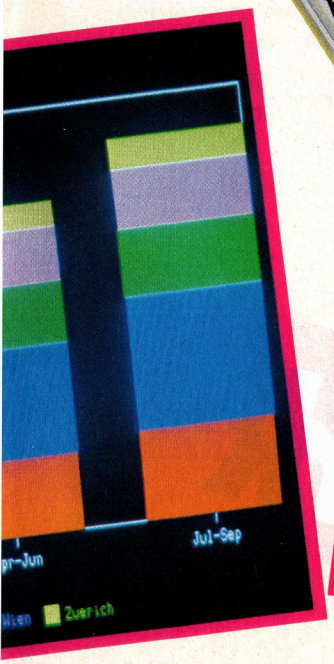
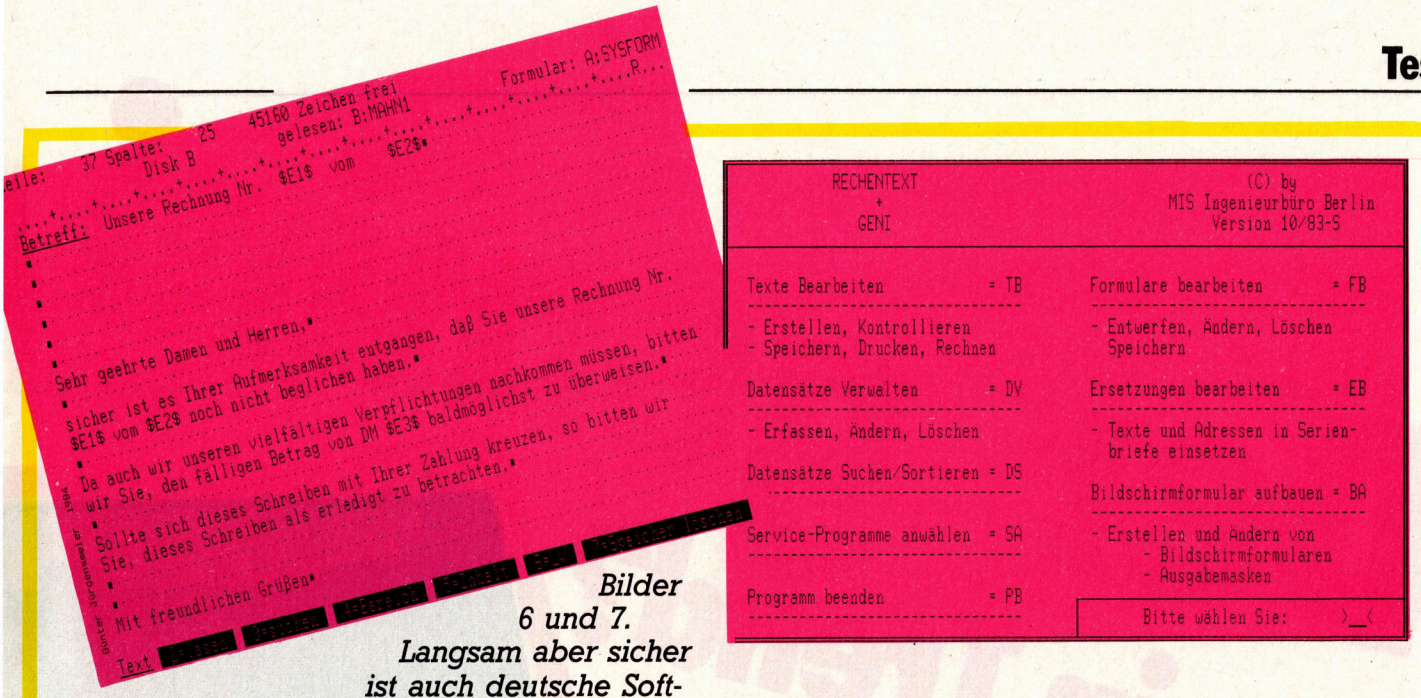


Bild 3. Die (amerikanische) Tastatur des TI-PPC, von ihm selbst auf dem hochauflösenden Farbmonitor gezeichnet.



Bilder 6 und 7. Langsam aber sicher ist auch deutsche Software für den TI-Computer erhältlich. Hier das Textprogramm Rechentext.

Auf dem monochromen Monitor werden die acht möglichen Farben mit unterschiedlichen Graustufen

Guter Bildschirm, brillante Farbgrafiken

angezeigt. Die hochauflösende Grafik gehört allerdings nicht zum Standard-Lieferumfang. Dafür muß die normale Bildschirmkarte erst mit dem erforderlichen Speicher erweitert werden. Es gibt dabei zwei Ausbaustufen: ein Modul mit einer Ebene (schwarz und eine von acht Farben) und ein Modul mit drei Grafikebenen (acht Farben). Diese Grafikerweiterungen belegen keinen eigenen Speicherplatz, sondern passen auf den Bildschirmadapter.

Der TI PPC kann, genau wie der TI PC, zusammen mit den Bürocomputer-Systemen BS300, BS600 und BS800 von Texas Instruments eingesetzt werden. Es wird dabei das Terminal 931 emuliert; zusätzlich sorgen Routinen im Betriebssystem der BS-Rechner für eine Verknüpfung der Datenbestände und lassen den wechselseitigen Zugriff auf die unterschiedlichen Massenspeicher zu.

Bei den Betriebssystemen hat man die freie Auswahl. Hauptbetriebssystem ist natürlich MS-DOS. Zur Zeit wird noch die Version 1.25 ausgeliefert, für diesen Test stand aber auch schon eine Vorabversion der Version 2.11 zur Verfügung. Darüber hinaus sind erhältlich: CP/M-86, Concurrent-CP/M-86 und das UCSD-p-System.

Das Basic des TI PPC entspricht im wesentlichen dem Advanced Basic des IBM-Personal Computers in der Version 1 (auf anderen Rechnern auch unter dem Namen GW-Basic bekannt). Befehle, die auf dem TI keinen Sinn machen (zum Beispiel MOTOR), produzieren eine Fehlermeldung. Bei den Grafikbefehlen gibt es die Erweiterung PALETTE, mit der es möglich ist, die Farbzuordnung schnell zu ändern. Das erlaubt ein einfaches Programmieren von sich bewegenden Objekten. Solange die Programme keine Grafikmöglichkeiten nutzen, ist die Umstellung von IBM-Basic-Programmen sehr einfach; es ist dabei lediglich zu beachten, daß der TI 12 Funktionstasten hat und deshalb die Parameter bei einigen Varianten des KEY-Befehls anders aussehen. Sobald jedoch Grafikbefehle verwendet werden, müssen diese Programmteile aufgrund der wesentlich höheren Auflösung des TI PPC überarbeitet werden. Allerdings sind die Grafikbefehle auf dem TI einfacher zu handhaben, da man es hier nicht mit einer Vielzahl von verschiedenen Grafikmodi zu tun hat, sondern nur mit der einen Auflösung von 720 x 350 Punkten und acht Farben (320 x 200 Punte mit vier Farben oder 640 x 200 Punkte schwarzweiß auf dem IBM-PC). Zum Basic-Interpreter ist auch ein Compiler erhältlich.

Viele amerikanische Standardpakete sind auf den TI Professional umgesetzt worden und laufen ebenso auf dem Portable Professional Computer (zum Beispiel Multiplan und dBase II). Besonders schön ist die Lotus 1-2-3-Version, denn es werden die Grafikmöglichkeiten voll genutzt (Bilder 4 und 5). Das Angebot an deutscher Software ist noch gering, mit Unterstützung von TI sind jedoch zahlreiche Softwarehäuser dabei, ihre Programme an den Professional und Portable Professional anzupassen. An Textverarbeitungen werden zum Beispiel in Kürze gleich zwei deutsche Programme erhältlich sein, TEX-ASS und Rechentext (Bilder 6 und 7). Als Drucker kann dabei der TI 855 eingesetzt werden (Bild 8). Das integrierte Paket OPEN ACCESS soll in einer deutschen Version zur Hannover-Messe vorgestellt werden.

Der Portable Professional Computer wird ab etwa Juli/August lieferbar sein und zwar in einer voll eingedeutschten Version (deutsche Tastatur, deutsches MS-DOS und so weiter).

Der TI Portable Professional Computer ist ein System, das zwar oberflächlich betrachtet auf der Welle der IBM-Kompatiblen zu schwimmen scheint, sich aber in wesentlichen Punkten von solchen Computern abhebt. Vorteile des TI-Portable Computers gegenüber seinem Hauptkonkurrenten, dem IBM-Personal Computer, sind die bessere Tastatur, die hochauflösende Grafik und die besseren »Aufstiegsmöglichkeiten«. Während man beim IBM-PC keine Möglichkeit hat, aus einem PC ohne Erweiterungseinheiten einen XT zu machen, ist bei TI der Aufstieg von einem normalen Professional oder Portable Professional zu einem mit Festplatte ohne Problem möglich.

Im großen und ganzen ist der TI PPC ein System, das man fast ohne Kritik als »gut« einstufen kann.

(jü)

Wanderdisco modern

Stimmung

Auf dem Lande oder in einer Kleinstadt zu leben hat unbestritten viele Vorteile, aber ein Teenager, der Abwechslung sucht, sieht die »Provinz« nicht immer in erster Linie positiv. Die Bedürfnisse dieser Jugendlichen nutzen professionelle Wanderdiscos: Mal hier, mal dort bieten sie Tanzveranstaltungen an und bringen mit Musik, Licht- und sonstigen Effekten das gewünschte sogenannte »Leben in die Bude«.

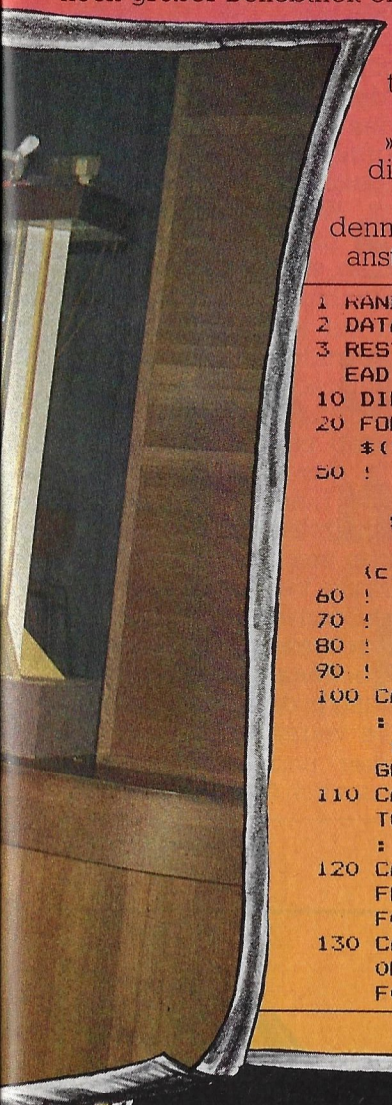
Wanderdisco: Für spezielle Effekte sorgt ein TI 99/4A

Mit dem Angebot der ab und zu in Hagenbach — einem etwa 5000 Einwohner großen Ort in der Nähe von Karlsruhe — anreisenden Discotheken war eine Gruppe einheimischer Jugendlicher überhaupt nicht mehr einverstanden: Es wurde ihnen einfach zu »kommerziell«, die Musik entsprach nicht ihrem Geschmack und war dazu noch zu einseitig »auf den üblichen Discosound ausgerichtet«. Das ganze Drumherum allein war das Eintrittsgeld auch nicht wert. Anstatt sich darüber nur aufzuregen, beschloß man, etwas zu tun: 1980 gründeten einige der Jugendlichen ihre eigene »Wanderdisco« und nannten sie wegen der Vorliebe für die gleichnamige Musikrichtung »Rock-Disco«. Eigeninitiative, unermüdlicher Einsatz, Ausdauer, Begeisterung und das Bemühen, die Wünsche der Jugendlichen und nicht

— Der Computer sorgt für

kommerzielle Interessen in den Vordergrund zu stellen, sorgten dafür, daß diese Wanderdisco sich heute — nach vier Jahren — immer noch großer Beliebtheit erfreut.

Genau betrachtet ist der Begriff »Wanderdisco« irreführend, denn die Veranstaltungen finden immer in der Mehrzweckhalle der Gemeinde Hagenbach statt. Es handelt sich eher um eine »stationäre Wanderdisco«. In den Gründerzeiten gab es einige Probleme, die Idee war zwar da, aber keine Quelle für finanzielle Mittel, um das Ganze auch mit einem einigermaßen professionellen Außenanstrich aufzuziehen. Nach dem Motto »mitgefangen, mitgehangen« bissen die Organisatoren in den sauren Apfel und stellten ihre privaten Musikanlagen zur Verfügung. Das kostete



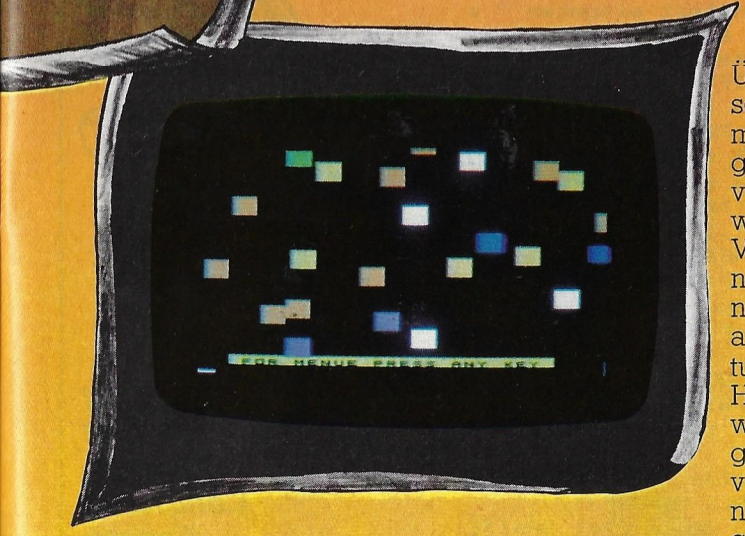
```
1 RANDOMIZE
2 DATA TITEL.MUSTER.JGH.RANDOM
3 RESTORE 2 :: FOR I=1 TO 5 :: R
  EAD MENUE$(I):: NEXT I
10 DIM Z$(20)
20 FOR I=1 TO 3 :: CALL ZOOM(SEG
  $("JGH",I,1),Z$(I)):: NEXT I
50 !   GERD HERRMANN

     SOFTWAE LABEL

   (c) by G.H 3.83
60 !
70 !
80 !
90 !
100 CALL CLEAR :: FOR I=1 TO 9 :
    : CALL COLOR(I,8,2):: NEXT I
     :: CALL SCREEN(2):: CALL MA
    GNIFY(4)
110 CALL COLOR(3,2,8):: FOR I=9
    TO 14 :: CALL COLOR(I,2,12):
    : NEXT I
120 CALL CHAR(140,"FOFOFOFOFOFOF
    FFFFFFFFOFOFOFOFOOFOFOFOFO
    FOFFFFFFFFFOFOFOFOFOFOF")
130 CALL CHAR(136,"1F7F7FFFF8FOF
    OFOFOFOFOF8FF7F7F1FF8FEFEFF1
    FOOOO3C3EOFOF1FFFFEFEF8")
```

TI 99/4A

Basic-Listing für den »Disco-Computer«



Lichteffekte wie diese sehen »laufend« natürlich viel besser aus

Überwindung, denn schließlich wußte man nie vorher so genau, wie man es verkraften würde, wenn plötzlich der Verstärker oder eine Lautsprecherbox nach einer Tanzveranstaltung reparaturbedürftig nach Hause getragen werden mußte. Abgesehen davon: Den von den professionellen Wanderdiscos gewohnten stimmungsanregenden »Schnickschnack«, wie Lichteffekte gab es nur als stümperhafte Andeutung. Trotzdem, der rege Zulauf der Jugendlichen sorgte für den nötigen Mut, weiterzumachen.

Für Gerd Herrmann, einen Mitbegründer der neuen Disco, war es nichts Neues, ein eher gemeinnütziges Unterfangen für Jugendliche zu unterstützen. Er hatte schon lange vorher in der Selbstverwaltung des örtlichen Jugendheims mitgewirkt. Finanzielle und andere Probleme sowie deren Lösung durch Eigeninitiative und Engagement waren ihm schon seit längerem vertraut. In seiner Rückschau dauerte die »Durststrecke« zwei Jahre — dann ging's richtig bergauf. Anstelle der Gelder, die zunächst aus Privatinitiative zugebuttert wurden, konnte man nach den ersten zwei "mageren Jahren" Anschaffungen tätigen, die sich aus Einnahmen der Eintrittsgelder finanzierten. Anfangs waren es drei Mark pro Nase. Nach und nach wurde eine Musikanlage aufgebaut, die ausschließlich für Discozwecke reserviert war. Das Do-it-yourself-Prinzip wurde — wo immer möglich — durchgehalten.

Auftritt mit BAP

Die Hagenbacher bleiben aber nicht nur »unter sich«. Mit Stolz denken sie noch oft an ihren aufsehenerregenden Auftritt mit dem Südwestfunk und der Kölner Rockgruppe BAP zurück. Dabei werden sie das Gefühl nicht los, daß dieses Ereignis enorm zum Bekanntwerden von BAP beigetragen hat.

Mittlerweile strebt man nach einem immer weitergehenden technischen Perfektionismus: In einigen Wochen soll ein 16-Kanal-Computerlichtsteuergerät in Betrieb genommen werden. Diesem Ausbau liegt allerdings momentan noch ein dicker Stein im Weg: Die Stromversorgung in der Hagenbacher Mehrzweckhalle ist mit zirka 300000 Watt Höchstleistung nicht für derartige Zwecke ausgelegt. Diese Beschränkung macht sich manchmal schon recht »finster» bemerkbar. Es muß vorher genau kalkuliert werden, ob das derzeitige Flaggschiff der Lichtanlage, ein vier Meter hoher Stern,

Die Anlage wird ständig erweitert — natürlich im Do-it-yourself-Verfahren

der aus acht Segmenten besteht — natürlich selbstgebaut — und etwa 7000 Watt Lichtleistung liefert (und verbraucht) eingeschaltet werden kann. Er könnte ja den Computer stören ....

Der Computer darf nie fehlen

Der Computer, ein TI 99/4A, spielt eine entscheidende Rolle bei der Hagenbacher Rockdisco. Anfangs wurde er eingesetzt, um Ordnung in das fürchterliche Plattenchaos zu bringen. Schließlich stand kein professioneller Discjockey zur Verfügung; Wiederholungen von Platten oder bange Sekunden der Entscheidung, was jetzt wohl am besten zur Stimmung des Publikums paßte und auf welcher Langspielplatte sich gerade der entsprechende Song versteckt hielt, waren an der Tagesordnung. Gerd Herrmann schrieb ein Programm, das Ordnung in das anfängliche Wirrwarr brachte: Eine Art Dateiverwaltung für Schallplatten.

Alle Daten von Titeln, die für einen Abend zur Verfügung standen, wurden mit Stichworten, wie »Rock« oder »Swing« sowie mit den entsprechenden Quellenangaben vor jeder Veranstaltung in den Computer eingetippt. Sie waren abends im Arbeitsspeicher und der Disjockey brauchte nur noch den gewünschten

Fortsetzung auf Seite 22

```
140 CALL CHAR(96,"")
150 REM
160 CALL SPRITE(#1,136,12,20,100
    ,#2,140,14,37,110)
170 DISPLAY AT(13,6):"ADVENTURE
    SOFTWARE" :: DISPLAY AT(15,1
    1):"PRESENT"
180 CALL PAUSE(2000):: CALL DELS
    PRITE(ALL)
190 DISPLAY AT(1,1)ERASE ALLBEEP
    :"logon ?"
200 DISPLAY AT(1,9):"" :: ACCEPT
     AT(1,9)(UALPHA)SIZE(5):A$ :
    : IF A$="DISCO" THEN 230
210 DISPLAY AT(1,9):"falsches ke
    yword" :: CALL PAUSE(250)::
    GOTO 200
220 REM
230 REM
240 REM   MENUE
250 CALL CLEAR :: FOR I=1 TO 4 :
    : DISPLAY AT(4+I*3,10):I;MEN
    UE$(I):: NEXT I
260 DISPLAY AT(24,5):"`choose`
    by  number`"
270 CALL KEY(0,K,S):: IF S=0 OR
    K<49 OR K>53 THEN 270
280 ON K-48 GOSUB 300,360,600,65
    0
290 GOTO 250
300 CALL CLEAR :: CALL DELSPRITE
    (ALL):: DISPLAY AT(1,1):"`er
    wuenschten`text`eingeben`" :
    : DISPLAY AT(5,1):" ``maxima
    l``eine``zeile``"
310 CALL MAGNIFY(2):: F=2+(INT(R
    ND*7)+1)*2 :: ACCEPT AT(10,1
    )SIZE(28):X$
320 CALL CLEAR :: D=LEN(X$):: FO
    R I=1 TO D :: CALL SPRITE(#I
    ,ASC(SEG$(X$,I,1)),F,I*(20-I
    NT(D/2))-1,I*(INT(256/D))-4,
    -15,0):: NEXT I
330 W=INT(RND*7+2)*2 :: FOR I=1
    TO D :: CALL COLOR(#I,W):: C
    ALL KEY(0,K,S):: IF S<>0 THE
    N RETURN
335 NEXT I :: GOTO 330
355 REM     MUSTER
360 CALL CLEAR :: CALL DELSPRITE
    (ALL)
370 CALL CHAR(136,"FFFFFFFFFFFFF
    FFF"):: CALL MAGNIFY(2)
375 FOR I=1 TO 28 :: CALL SPRITE
    (#1,136,RND*14+2,RND*190+1,R
    ND*250+1,RND*40-20,RND*40-20
```

Basic-Listing für den »Disco-Computer« (Fortsetzung)

Wanderdisco modern — Der Computer sorgt für Stimmung

Fortsetzung von Seite 18

Titel einzugeben, um zu erfahren, welche Platte als nächste zu spielen war. Und damit am selben Abend keine Wiederholung vorkam, wurde das jeweilige Lied sofort aus dem Arbeitsspeicher gelöscht. Auch eine kleine Statistik konnte mitgeführt werden: So wußte man schon recht bald, daß »Foreigner 4« bei den Jugendlichen von allen Rock-Disco-Platten am besten ankam, denn 40 Prozent aller Lieder dieser Platte wurden bereits als Hit Nummer 1 gewählt.

Ein Handicap hatte die Plattenverwaltung mit dem Computer: Es mußte sehr viel Zeit und Arbeit investiert werden, die Lieder vor jeder Veranstaltung auszusuchen und die entsprechenden Daten einzugeben. Und wenn dann noch wegen Überlastung des Stromnetzes die notwendige Energie während der Veranstaltung plötzlich wegblieb, war auch die mühevolle Eingabe der Plattentitel umsonst gewesen. Zu guter Letzt weigerte sich jeder, der schon einmal eine solche »Panne« miterlebt hatte, noch Titel einzugeben.

Heute gibt's Effekte vom Computer

Heute wird der TI 99 zur Erzeugung der sogenannten »Bühneneffekte« eingesetzt: Der Titel des gespielten Liedes wandert in Laufschrift über zwei große Monitore, die am Bühnenrand aufgestellt sind. Vor zwei Jahren hat Gerd Herrmann diesen »Gag« bei den Hagenbacher Tanzveranstaltungen eingeführt und seitdem mögen die Jugendlichen nicht mehr darauf verzichten. Aber nicht nur Laufschriften, sondern auch Grafiken, die in verschiedenen Farben über den Bildschirm laufen, werden von dem Programm erzeugt. Nach Ansicht des Autors läßt sich das Programm auch hervorragend »zweckentfremden«, indem es beispielsweise für die Gestaltung des Vorspanns bei Videofilmen benutzt wird.

```
    ):: NEXT I
380 FOR L=1 TO 15 :: FOR I=1 TO
    5 :: G=I*5 :: F=8+INT(RND*5)
    *2 :: CALL COLOR(#G-4,F,#G-3
    ,F,#G-2,F,#G-1,F,#G,F):: CAL
    L KEY(0,K,S):: IF S<>0 THEN
    RETURN
390 NEXT I :: NEXT L :: RETURN
600 CALL CLEAR :: CALL DELSPRITE
    (ALL):: CALL MAGNIFY(4)
610 FOR I=1 TO 3 :: CALL CHAR(12
    4+I*4,Z$(I)):: NEXT I
620 FOR I=1 TO 3 :: CALL SPRITE(
    #I,124+I*4,4,I*40,I*50):: NE
    XT I :: FOR I=4 TO 6 :: CALL
     SPRITE(#I,140-(I-3)*4,4,280
    -I*40,400-50*I):: NEXT I
630 FOR I=1 TO 5 :: CALL MOT ::
    FOR L=1 TO 25 :: CALL COLOR(
    #RND*6+1,INT(RND*7+2)*2):: C
    ALL KEY(0,K,S):: IF S<>0 THE
    N RETURN
640 NEXT L :: NEXT I :: RETURN
650 ON INT(RND*2)+1 GOSUB 360,60
    0
660 PRINT " `for`menue`press`any
    `key`",
670 FOR I=1 TO 500 :: CALL KEY(0
    ,K,S):: IF S<>0 THEN RETURN
680 NEXT I
690 FOR I=1 TO 14 :: CALL SCREEN
    (I):: CALL PAUSE(10):: NEXT
    I :: CALL SCREEN(2):: GOTO 6
    50
700 SUB ZOOM(A$,Z$):: IF FLG>0 T
    HEN 740
710 DIM I$(16)
720 DATA 00,03,0C,0F,30,33,3C,3F
    ,C0,C3,CC,CF,F0,F3,FC,FF
730 RESTORE 720 :: FOR I=1 TO 16
     :: READ I$(I):: NEXT I :: F
    LG=1
740 CALL CHARPAT(ASC(A$),X$)
750 LZ$,RZ$="" :: FOR U=1 TO 8 :
    : L=U*2-1 :: GOSUB 790
760 LZ$=LZ$&T$ :: L=U*2 :: GOSUB
     790
770 RZ$=RZ$&T$
780 NEXT U :: Z$=LZ$&RZ$ :: GOTO
     800
790 T$=RPT$(I$(POS("0123456789AB
    CDEF",SEG$(X$,L,1),1)),2):: 
    RETURN
800 SUBEND
810 SUB PAUSE(X):: FOR I=1 TO X
    :: NEXT I :: SUBEND
820 SUB MOT :: E=RND*60-30 :: X=
    RND*60-30 :: FOR I=1 TO 3 ::
     CALL MOTION(#I,E,X,#I+3,-E,
    -X):: NEXT I :: SUBEND
```

Basic-Listing für den »Disco-Computer« (Schluß)

Das Programm ist recht einfach in der Handhabung. Man benötigt einen TI 99/4A mit dem Extended Basic Modul und ein Kassettenlaufwerk. Nach dem Laden wird das Programm wie gewohnt mit RUN gestartet. Auf dem Bildschirm erscheint das Software — Label und nach einer kurzen Pause wird ein Code-Wort abgefragt. Dies dient zum einen dazu, einen vorzeitigen Programmablauf zu verhindern — bevor die Platte überhaupt aufgelegt ist und deren Titel am Bildschirm gezeigt wird. Zum anderen ist es auch als ein Teil des doch recht beliebten »Schnickschnacks« gedacht — der jeden erfreut, der während der Disco-Time den Computer bedienen darf. Das einzugebende Paßwort lautet »standardmäßig« DISCO. Nun erscheint das Hauptmenü:

1 Titel
2 Muster
3 Logo
4 Random

Zu 1: Nach Anwahl des Menüpunktes kann man eine Zeile beliebigen Textes einschließlich Zahlen und Sonderzeichen eingeben. Dieser Text läuft dann nach der Eingabebestätigung durch ENTER so lange über den Bildschirm, bis eine beliebige Taste gedrückt wird.

Zu 2: Der Bildschirm wird mit bunten, sich bewegenden Quadraten bestückt (siehe Bild 3), nach einiger Zeit erfolgt ein Rücksprung ins Hauptmenü.

Zu 3: Hier können eigene Logos ins Programm eingegeben werden, die wie die Laufschrift unter Menüpunkt 1 über den Bildschirm laufen. Standardmäßig ist die Buchstabenfolge »JGH« (Jugend-Gemeinschaft-Hagenbach) vorgegeben.

Zu 4: In zufälliger Reihenfolge wird mal das Untermenü 2 und mal das Untermenü 3 automatisch abgespielt, bis es durch Drücken einer beliebigen Taste abgebrochen wird. (Gerd Herrmann/kg)

Laden mehrerer Programme

Gelegentlich möchte man ein oder mehrere Programme im Hauptspeicher anhängen oder zusammenfügen. Der einfachste Weg ist der Kauf des Programmierhilfe-Moduls VC 1212. Es geht aber auch billiger.

Will man nämlich neben dem VC 1212 noch den Speicher erweitern, so benötigt man zusätzlich mindestens eine dreifache Erweiterungs-Box. Eine weitere Möglichkeit besteht mit dem Programm Merge/V3 aus CP 7/83. Es geht aber auch noch einfacher.

Programm 1 laden (im Direktmodus)

PRINT PEEK(43), PEEK(44) (die Werte notieren)
PRINT PEEK(45), PEEK(46)
POKE 43, Wert von 45 minus 2
POKE 44, Wert von 46

Programm 2 laden

POKE 43, alten Wert einsetzen
POKE 44, alten Wert einsetzen

Nun ist das Programm 2 an Programm 1 angehängt.

Wenn keine Sprungadressen vorhanden sind, läuft das neue Programm, auch wenn im Programm 1 die gleichen oder sogar höhere Zeilennummern vorhanden sind als in Programm 2.

Wenn Sprungadressen vorhanden sind, so ist darauf zu achten, daß im Programm 2 die Zeilennummern höher sind.

Wenn PEEK(45) = 1 ergibt, dann ist der neue Wert = 255 und PEEK(46) minus 1

Wenn ein Super-Erweiterungs-Modul gleichzeitig angeschlossen ist, so ist darauf zu achten, daß bei jedem Programm das Modul mit dem Befehl GRAPHIC 0 ausgeschaltet wird.

Auf diese Weise lassen sich beliebig viele Programme zusammenhängen, bis der Hauptspeicher voll ist.

(Willi Heusser)

Der TI 99/4A lernt laufen

Kleine Taschencomputer mit ihrer sehr begrenzten Anzeigenkapazität haben den Trick als erste benutzt um längere Texte ausgeben zu können: die Laufschrift. Aber auch der TI 99/4A mit Extended Basic kann Laufschrift erzeugen.

TI 99/4A

```
100 !****LAUFSCHRIFT****
110 L$="Dies ist ein Probetext" ! AUSGABETEXT
120 Z=10 ! AUSGABEZEILE
130 S=5 ! AUSGABESPALTE
140 L=10 ! AUSGABELAENGE
150 A=5 ! ANZAHL DER AUSGABEWIEDERHOLUNGEN
160 CALL LAUFSCHRIFT(L$,Z,S,L,A)!HIER STEHT DER AUFRUF IM HAUPTPROGRAMM
170 !
180 !
190 SUB LAUFSCHRIFT(L$,Z,S,L,A)
200 L$=RPT$(" ",L)&L$
210 FOR LO=0 TO A*LEN(L$):: DISPLAY AT(Z,S)SIZE(L):SEG$(L$,1,28):: L$=SEG$(L$,2,
100)&CHR$(ASC(L$)):: NEXT LO :: SUBEND
```

Auf dem Bildschirm des TI 99/4A mit seinen 32 Zeichen pro Zeile geht es schnell sehr eng zu, wenn etwas mehr Informationen gleichzeitig dargestellt werden sollen, vor allem, wenn diese aus längeren Texten bestehen. In dieser Situation entstand die Idee, dem Platzmangel mit einer Laufschrift abzuhelfen.

Mit dem TI-Basic muß dieser Gedanke allerdings ein Traum bleiben. Die dazu notwendigen Routinen sind viel zu langsam. In Extended Basic indessen ist die Laufschrift mit einigen wenigen Programmzeilen zu verwirklichen. Formuliert man dann diese Routine noch als Subroutine, kann die Laufschrift wie ein fest eingebauter Basic-Befehl mit »CALL LAUFSCHRIFT (L$,Z,S,L,A)« aufgerufen werden. Die Argumente stehen für den Ausgabetext, die Ausgabezeile und -spalte sowie die Ausgabelänge und Zahl der Wiederholungen. Statt der vorprogrammierten Zahl der Wiederholungen kann natürlich auch eine CALL KEY-Abfrage in die Subroutine eingebaut werden, welche die Ausgabe bei Betätigung einer beliebigen Taste abbricht.

Listing »Laufschrift« Die Subroutine (ab Zeile 190) sollte an das Ende eines Hauptprogramms gestellt werden.

Die eigentliche Subroutine ist lediglich drei Zeilen lang und findet somit in nahezu jedem Programm ihren Platz. Sie beginnt im Listing in Zeile 190. Sinnvollerweise sollte man allerdings die Routine an den Schluß des Hauptprogramms stellen. Die Werte in den Zeilen 100 bis 160 sind nur Beispiele. Ein Ausgabetext kann ohne weiteres auch länger als eine Zeile sein, nur ist dann der Effekt nicht mehr besonders angenehm.

Sinnvoll kann die Laufschrift auch im Rahmen von Ausgabemasken eingesetzt werden. Umrahmungen des Ausgabefeldes werden durch die Laufschrift nämlich nicht zerstört. Die Ausgabe-Geschwindigkeit kann bei Bedarf durch eine kleine Delay-Schleife jederzeit verlangsamt werden.

(Irmgard Maier)

Anschnallen, Commodore- und Texas Instruments-Freaks: ATARI POLE POSITION jagt Euch jetzt mit 280 km/h durchs Wohnzimmer.

Im Formel-1-Tempo erobern die Superstars von ATARI jetzt noch mehr Commodore- und TI-Fans. Mit ATARISOFT, dem Programm, das die erfolgreichsten Arkaden-Hits als Heimversion ins Wohnzimmer bringt.

So können Sie jetzt Ihren Commodore 64, Ihren Commodore VC 20 und Ihren TI 99/4A von Texas Instruments mit den spannenden Superhits von ATARI auf Hochtouren bringen.

Vorneweg POLE POSITION[1] mit schweißtreibenden Rennduellen auf heißer Asphaltpiste. Und CENTIPEDE[2], JUNGLE HUNT[3], PAC-MAN[4], DEFENDER[5], DONKEY KONG[6]*, MS. PAC-MAN[7], JOUST[8]*, DIG DUG[9], MOON PATROL[10], ROBOTRON: 2084[11]* und GALAXIAN[12]*. Allesamt voller Action. Allesamt erhältlich, wo es auch ATARI gibt. Und allesamt zum Verrücktwerden.

Wir informieren Sie gerne ausführlich und kostenlos über ATARISOFT. Schicken Sie dazu einen mit DM 0,80 frankierten und mit Ihrer Adresse versehenen DIN C 4-Umschlag an: ATARI, ATARI-Club, Postfach 60 01 68, 2000 Hamburg 60. Anschriften von ATARI-Händlern in Ihrer Nähe erfahren Sie telefonisch von unseren ATARI-Verkaufsbüros: Büro Nord, Hamburg, 040 / 2 20 10 11, Büro West, Ratingen, 02102 / 3 20 76 - 78, Büro Mitte, Eschborn, 06196 / 48 17 22/ 23. In der Schweiz: 8400 Winterthur, 052 / 23 62 31. In Österreich: 1210 Wien, 222 / 30 15 62.

* Donkey Kong: nicht für Commodore 64, Joust, Robotron: 2084, Galaxian: nicht für TI 99/4A.

ATARISOFT®

© 1984 Atari, Inc. All rights reserved. Änderungen vorbehalten.

[1] © Namco 1982. [2] © 1983 ATARI, Inc. [3] © Taito America Corporation. [4] © Namco 1980. [5] © Williams 1980. [6] © Nintendo 1981, 1983. [7] © Namco 1982. [8] © Williams 1982. [9] © Namco 1982. [10] © Williams 1982. [11] © Williams 1982. [12] © Namco America.

```
498   REM
499   REM ----- EINGABE TONLAENGE/PAUSENLAENGE -----
500   INPUT"GESCHWINDIGKEIT:

      -1-   40  BPM
      -2-   50  BPM
      -3-   60  BPM
      -4-   90  BPM
      -5-  140  BPM";G: G=5-G
510   W=G^(3/2)*64:POKE&HB001,INT(W/256)+1:POKE&HB003,((W/256)-
      INT(W/256))*128+127: POKE&HB021,G*50
597   REM
598   REM
599   REM ----- MENUE -----
600   CLS: PRINT@24,"MORSETRAINER": PRINTSTRING$(63,143): PRINT:
      INPUT"
      -1- TRAINING
      -2- UMSETZUNG
      -3- SIMULTAN
      -4- NEUE GESCHWINDIGKEIT";MODE
610   IF MODE>4 OR MODE<1 THEN600 ELSE IF MODE=4 THEN500
620   D$=INKEY$: IF MODE=3 THEN GOSUB2000 ELSE IF MODE=1 THEN3000
      ELSE IF MODE=2 THEN LINEINPUTA$: INPUT"WIEDERHOLUNGEN";RPT
997   REM
998   REM
999   REM ----- UMSETZUNG VON A$ IN MORSEZEICHEN -----
1000  IF A$="*" THEN600 ELSE FOR H=1 TO RPT: FOR K=1 TO LEN(A$)
1010  B$=MID$(A$,K,1)
1020  B=ASC(B$): B=B-43: IF B<1 OR B>47 THEN B$="": GOTO1040
1030  B$=A$(B)
1040  IF B$="" THEN FORJ=1 TO G*10: NEXTJ: GOTO1100
1050  FOR J=1 TO LEN(B$)
1060  ON VAL(MID$(B$,J,1)) GOTO1070,1080
1070  X=USR2(0): GOTO1090
1080  X=USR1(0)
1090  NEXTJ
1100  FOR J=1 TO (G-1)*30: NEXTJ
1105  IF MODE<>3 THEN NEXTK,H
1110  FOR J=1 TO (G-1)*30: IF AF=1 THEN RETURN ELSE620
1997  REM
1998  REM
1999  REM ----- SIMULTANUEBERSETZUNG -----
2000  A$=D$: PRINTD$;: IF D$="*" THEN500
2010  IF D$=CHR$(13) THEN600
2020  IF D$="" THEN A$=" "
2030  RETURN
2997  REM
2998  REM
2999  REM ----- AUSGABE VON BUCHSTABENGRUPPEN -----
3000  AF=1: CLS: PRINT@24,"MORSETRAINING": PRINTSTRING$(63,143)
3010  PRINT"ASCII-ZEICHENBEREICH:": INPUT"VON";A1: INPUT"BIS";A2:
      A2=A2+1
3015  IF A1<44 OR A2<44 OR A1>91 OR A2>91 OR A1>=A2 THEN3000
3020  FOR I=1 TO 10
3030  T$(I)="": FOR J=1 TO 5: T$(I)=T$(I)+CHR$(A1+RND(A2-A1)-1):
      NEXTJ,I
3040  A$="": FOR I=1 TO 10: A$=A$+T$(I)+" ": NEXTI: RPT=1
3050  PRINT"OK? DANN <ENTER>!"
3060  IF PEEK(14400)<>1 THEN3060 ELSE GOSUB1000
3070  CLS: FOR I=1 TO 10
```

```
3080  PRINT"GRUPPE"I;: INPUTD$(I): NEXTI
3090  CLS: K=0: PRINT"GRUPPE","GEGEBEN","EMPFANGEN","FEHLER"
3100  F=0: FORI=1TO10: PRINTI,T$(I),D$(I),: S=5: FOR J=1 TO 5: IF
      MID$(T$(I),J,1)=MID$(D$(I),J,1) THEN S=S-1
3110  NEXTJ: PRINTS :F=F+S: NEXTI
3120  PRINT"FEHLER GESAMT:"F: INPUT"-1- WEITER ODER -2- MENUE";P
3130  IF P=1 THEN3000 ELSE AF=0: GOTO600
9997  REM
9998  REM
9999  REM ----- SOUND-ROUTINE -----
10000 DATA 22, 1, 6, 255, 62, 1, 205, 23, 176, 62, 0, 205, 23, 176,
      5, 194, 4, 176, 21, 194, 2, 176, 201, 211, 255, 14, 48, 13,
      194, 27, 176, 201, 6, 95, 22, 1, 195, 4, 176
10098 REM
10099 REM ----- ZEICHENFOLGEN FUER ASCII-CODES 43-90 -----
11000  DATA"221122", "21112", "121212", "21121", "22222", "12222"
11010  DATA"11222", "11122", "11112", "11111", "21111", "22111"
11020  DATA"22211", "22221", "222111", "", "", ""
11030  DATA"", "112211", "", "12", "2111", "2121"
11040  DATA"211", "1", "1121", "221", "1111", "11"
11050  DATA"1222", "212", "1211", "22", "21", "222"
11060  DATA"1221", "2212", "121", "111", "2", "112"
11070  DATA"1112", "122", "2112", "2122", "2211"
```

ANM: Ä = EXPONENT (HOCHPFEIL)
ß = KLAMMERAFFE

Listing zu »Morsetrainer« (Schluß)

Miniroutine: Mehrwertsteuer-berechnung

Manchmal fehlt nur eine kleine Miniroutine, um ein Stündchen Arbeit zu sparen. Verwenden wir für solche Zwecke ruhig mal unseren Heimcomputer. Mit der hier präsentierten Miniroutine können Sie auf Ihrem TI 99/4A jeden beliebigen Endbetrag auf den Ausgangsbetrag (ohne Mehrwertsteuer) und den Mehrwertsteuer-Anteil zurückrechnen. Oder Sie bauen den Algorithmus (Zeile 130 und 140) in Ihr eigenes Programm ein. (lg)

```
100 CALL CLEAR
110 INPUT "BETRAG? ":B
120 INPUT "Mwst-Satz (in %)?
    ":MWSTS
130 AB=B/(1+MWSTS/100)
140 MWST=B-AB
150 PRINT "Ausgangsbetrag: "
    ;AB
160 PRINT "Mwst-Anteil ist:"
    ;MWST: :
170 GOTO 110
```

TI 99/4A

Listing für Mehrwertsteuer-Berechnung

Gesellschaft für Unternehmensberatung und Datenverarbeitung mbH

Andreas-Herz-Straße 4a
8011 Baldham b. München
Telefon 08106/7693

● Stellen Sie Ansprüche. Vergleichen Sie die Preise! Was Sie bei anderen PCs extra bezahlen müssen, ist beim Ibm-kompatiblen, preisgünstigen Corona PC serienmäßig bereits eingebaut: z.B. der monochrome Bildschirmadapter. Oder die Anschlüsse für Drucker und der große Arbeitsspeicher (128.000 Zeichen!). Oder die vier freien Adapter für Erweiterungen nach Ihren individuellen Wünschen.

● Von großem Vorteil für Ihren Arbeitsalltag: die **volle** technische Kompatibilität. Und Sie können **alle** professionellen PC-Programme einsetzen. (Wir liefern Ihnen jede marktgängige Software!)

● Für Einsteiger besonders wichtig: Der Profi-Computer Corona PC sagt Ihnen, wo's lang geht. Ein anschauliches Trainingsprogramm macht Sie zum Computer-Profi.

● Wenn Sie viel auf Reisen oder in Konferenzen sind: der Corona PC als tragbares Gerät. Einer der wenigen mit Festplatte!

● Wollen Sie wissen, was der Corona PC wirklich wert ist? Dann stellen Sie fest, für was Sie bei anderen alles Aufpreise bezahlen müssen.

● Serviceleistung: kostenlose Beratung und Unterstützung auch nach dem Kauf

corona Die professionellen Personal Computer

Kapitän auf hoher See

Mit dem nachfolgenden Spiel »Regatta« für den TI 99/4A (mit extended Basic) können sich zwei Freizeitkapitäne im Segeln üben.

Das Spiel »Regatta« ermöglicht das Erlernen des taktisch richtigen Segelns einer Kreuz, der wichtigsten Bahn in einer Regatta. Zwei Boote gehen nach einem Startsignal mit Backbordbug über die Startlinie. Die Startlinie ist in Lee soweit vorgezogen, daß das Leeboot mit dem Luvboot chancengleich ist. Die Boote laufen immer unter 45 Grad zur Hauptrichtung. Die zwei Spieler können über die geteilte Tastatur oder über Joystick ihr Boot zur Wende veranlassen. Bei Begegnung der beiden Boote muß die Grundregel, Backbord vor Steuerbord, eingehalten werden sonst wird das die Vorfahrt verletzende Boot hinter die Startlinie zurückgesetzt. Die Ziel-Bojen dürfen nicht berührt werden, sonst wird das Boot um einige Bootslängen zurückgesetzt.

(Werner Stresemann)

Listing zum Spiel »Kapitän auf hoher See«

```
100 REM REGATTASPIEL VON        WER
    NER STRESEMANN       GATOWERSTR
    .83C BERLIN 20
110 DISPLAY AT(2,10)BEEPERASE ALL:"
    REGATTASPIEL"
120 DISPLAY AT(6,1):"BOOT #1(LEE)"
130 DISPLAY AT(8,1):"TASTE  BUG","
        ","S    BACKBORD","D    STE
    UERBORD","X    VORWIND"
140 DISPLAY AT(13,1):"BOOT #2(LUV)"
150 DISPLAY AT(15,1):"TASTE    BUG"
    ,"      ","J    BACKBORD","K
    STEUERBORD","M    VORWIND"
160 DISPLAY AT(22,5):"DRUECKEN SIE *
    ENTER*    FUER SPIELBEGINN"
170 ACCEPT AT(24,15):B$
500 CALL CLEAR :: CALL SOUND(3000,1
    10,0,400,2):: CALL SCREEN(8)
510 RANDOMIZE
600 CALL CHAR(116,"040E1F1B170F3F5F
    9F0E0000000000000000000000000000
    000")
610 CALL CHAR(120,"F0F8DCDE5F3F3E2C
    2020000000000000000000000000000
    000")
620 CALL CHAR(124,"3F7FEF7338020100
    0000000000000080C0C0C0800000080
    000")
630 CALL CHAR(128,"70F8F8F8F8F8C7F8
    7020000000000000000000000000000
    000")
640 CALL CHAR(100,"7FFFFDFB77102040
    0000000000000000080C0800000000
    000")
650 CALL CHAR(104,"0F1F3B7BFAFC7C34
    0404000000000000000000000000000
    000")
660 CALL CHAR(108,"2070F8D8E8F0FCFA
    F9700000000000000000000000000000
    000")
670 CALL CHAR(112,"0E1F1F1F1F1FE31F
    0A0400000000000000000000000000
    000")
1000 CALL MAGNIFY(3)
1050 CALL CHAR(96,"1F1F1F10107C7C7C
    0000000000000000000000000000000
    0000")
1060 CALL CHAR(140,"081C3E7FFF7F3F1
    F0F0703010103070F0000000081C3E
    7FFFFFFFFFFFFFFFFF")
1070 CALL CHAR(136,"070707070707070
    7FF7F3F1F0F070301E0E0E0E0E0E0E
    0E0FFFEFCF8F0E0C080")
1080 CALL CHAR(132,"0000000081C3E7F
    FFFFFFFFFFFFFFFF10387CFEFFFFF
    CF8F0E0C08080C0E0F0")
1100 CALL CLEAR :: CALL SPRITE(#4,9
    6,7,16,100,0,0,#5,96,7,16,160,
    0,0)
1110 CALL SPRITE(#6,96,7,162,64,0,0
    ,#7,96,7,170,140,0,0)
1170 B=136 :: C=120 :: D=120 :: X=0
     :: Y=0 :: U=0 :: V=0
1200 CALL SPRITE(#1,C,14,174,90,X,Y
    ,#2,D,2,180,128,U,V,#3,B,13,10
    0,100,0,0)
1210 FOR SEC=10 TO 0 STEP -1
1220 DISPLAY AT(22,1)SIZE(3):SEC
1230 FOR DELAY=1 TO 100 :: NEXT DEL
    AY :: NEXT SEC
1240 DISPLAY AT(22,1)SIZE(3):
```


Die Drei

Preise inklusive gesetzlicher Mehrwertsteuer.
Versand erfolgt per Nachnahme oder frei Haus bei Vorkasse.

Personal Computer

NCR Decision Mate V und Osborne

Zubehör

EDV-Papier 2000 Blatt / A4		
für GP 100/250/550/700	DM	47,60
für GP 80	DM	58,50
Farbband		
für GP 80/100/250/550	DM	23,50
für GP 700 A	DM	48,-
Farb-Refill für GP 700 A	DM	32,90

Drucker

SEIKOSHA MATRIX DRUCKER		
GP-50A	DM	437,-
GP-100A	DM	747,-
GP-100AS (2K Buffer)	DM	847,-
GP-250X	DM	895,-
GP-550A	DM	948,-
INTERFACE-Preise für anschlußfertige SEIKOSHA DRUCKER außer GP 100 AS		
an TI99/4A, (Konsole)	DM	250,-
an SINCLAIR-ZX81	DM	140,-
an SINCLAIR-SPECTRUM	DM	180,-
an apple IIe	DM	250,-
an DRAGON 32	DM	80,-
an SHARP MZ 700	DM	250,-
an COMMODORE VC20/C64	DM	250,-

Angebot der Woche

OSBORNE 1DQD incl. CP/M, dt. Tastatur, 80 Zeichenkarte	DM	4999,-
MS-15 Typenraddrucker V24 und Centronics-Interface	DM	1498,-
MS-15 + Commodore-Interface	DM	1698,-
GP 700 A Farbgraphikdrucker 7 Grundfarben, Centr. Interf.	DM	1435,-
GP-550A für IBM PC	DM	999,-
TECO Bildschirm	DM	349,-
GP-100VC (VC20 + C64)	DM	596,-
GP-250X für SHARP MZ80A	DM	1154,-

Wir schließen auch alle Geräte gegen Berechnung an und weisen Sie ein; z.B. Einweisung in Wordstar DM 390,00.

ACHTEN SIE AUF UNSERE NÄCHSTEN ANZEIGEN

Computer-Versandhäuser

H. Terhechte-Abels
Natz-Thierweg 56
4400 Münster

U. Beck
Boosstraße 2
8000 München
☎ 089-652392

U. Hellmann
Eichkatzweg 10
3107 Ovelgönne
☎ 05084-4369

Listing: »Kapitän auf hoher See« (Schluß)

TI 99/4A

```
1245 A=2 :: CALL SOUND(100,150,0)::
     GOTO 1270
1250 A=INT(3*RND+1):: W=4*A+128 ::
     IF ABS(W-B)>4 OR W-B=0 THEN 12
     50 ELSE 1260
1260 B=W :: CALL PATTERN(#3,B):: CA
     LL SOUND(1000,-6,4)
1270 ON A GOSUB 3000,3100,3200
1275 BOE=INT(6*RND)+3
1280 FOR DELAY=1 TO BOE
1500 CALL KEY(1,K,@)
1520 IF @=0 OR K>3 THEN 1650
1530 IF K=0 THEN K=1
1540 ON K GOTO 1550,1560,1590
1550 IF C<112 THEN C=112 ELSE C=128
      :: X=3 :: Y=0 :: GOSUB 3450
1555 GOTO 1650
1560 IF A=1 THEN GOSUB 3300
1570 IF A=2 THEN GOSUB 3310
1580 IF A=3 THEN GOSUB 3320
1585 GOTO 1650
1590 IF A=1 THEN GOSUB 3330
1600 IF A=2 THEN GOSUB 3340
1610 IF A=3 THEN GOSUB 3350
1650 CALL KEY(2,L,S)
1660 IF S=0 OR L>3 THEN 1800
1665 IF L=0 THEN L=1
1670 ON L GOTO 1680,1690,1730
1680 IF D<112 THEN D=128 ELSE D=112
      :: U=3 :: V=0 :: GOSUB 3460
1685 GOTO 1800
1690 IF A=1 THEN GOSUB 3360
1700 IF A=2 THEN GOSUB 3370
1710 IF A=3 THEN GOSUB 3380
1720 GOTO 1800
1730 IF A=1 THEN GOSUB 3390
1740 IF A=2 THEN GOSUB 3400
1750 IF A=3 THEN GOSUB 3410
1800 REM VORFAHRT
1810 CALL COINC(#1,#2,5,H)
1820 IF H=0 THEN 1850
1830 IF C>112 THEN CALL DELSPRITE(#
     2):: CALL SOUND(200,110,0,300,
     0):: GOTO 1860
1840 CALL DELSPRITE(#1):: CALL SOUN
     D(200,110,0,300,0)
1850 REM BOJE
1860 CALL COINC(#1,#4,7,I):: CALL C
     OINC(#1,#5,7,IE):: IF I=0 AND
     IE=0 THEN 1870 ELSE CALL LOCAT
     E(#1,50,130):: CALL SOUND(200,
     130,0)
1870 CALL COINC(#2,#4,7,J):: CALL C
     OINC(#2,#5,7,JE):: IF J=0 AND
     JE=0 THEN 1900 ELSE CALL LOCAT
     E(#2,50,130):: CALL SOUND(200,
     130,0)
1890 REM  ZIEL
1900 CALL COINC(#1,1,130,20,Z):: IF
      Z=-1 THEN BT=1 :: GOTO 3500
1920 CALL COINC(#2,1,130,20,Z):: IF
      Z=-1 THEN BT=2 :: GOTO 3500
2800 NEXT DELAY
2900 GOTO 1250
3000 REM 45 GRAD
3010 IF C>112 THEN GOSUB 3300
3020 IF C<=112 THEN GOSUB 3330
3030 IF D>112 THEN GOSUB 3360
3040 IF D<=112 THEN GOSUB 3390
3050 RETURN
3100 REM 90 GRAD
3110 IF C>112 THEN GOSUB 3310
3120 IF C<=112 THEN GOSUB 3340
3130 IF D>112 THEN GOSUB 3370
3140 IF D<=112 THEN GOSUB 3400
3150 RETURN
3200 REM 135 GRAD
3210 IF C>112 THEN GOSUB 3320
3220 IF C<=112 THEN GOSUB 3350
3230 IF D>112 THEN GOSUB 3380
3240 IF D<=112 THEN GOSUB 3410
3250 RETURN
3300 C=116 :: X=-1 :: Y=0 :: GOTO 3
     450
3310 C=120 :: X=-1 :: Y=-1 :: GOTO
     3450
3320 C=124 :: X=0 :: Y=-1 :: GOTO 3
     450
3330 C=100 :: X=0 :: Y=+1 :: GOTO 3
     450
3340 C=104 :: X=-1 :: Y=1 :: GOTO 3
     450
3350 C=108 :: X=-1 :: Y=0 :: GOTO 3
     450
3360 D=116 :: U=-1 :: V=0 :: GOTO 3
     460
3370 D=120 :: U=-1 :: V=-1 :: GOTO
     3460
3380 D=124 :: U=0 :: V=-1 :: GOTO 3
     460
3390 D=100 :: U=0 :: V=1 :: GOTO 34
     60
3400 D=104 :: U=-1 :: V=1 :: GOTO 3
     460
3410 D=108 :: U=-1 :: V=0 :: GOTO 3
     460
3450 CALL PATTERN(#1,C):: CALL MOTI
     ON(#1,X,Y):: RETURN
3460 CALL PATTERN(#2,D):: CALL MOTI
     ON(#2,U,V):: RETURN
3500 CALL DELSPRITE(ALL):: DISPLAY
     AT(3,5)BEEPERASE ALL:"SIEGER :
     BOOT #";BT :: CALL SOUND(3000,
     110,0,400,0)
3510 DISPLAY AT(10,5):"NOCH EIN STA
     RT ?","DRUECKEN SIE BITTE=: *EN
     TER*","SONST EINE BELIEBIGE TA
     STE  FUER SPIELENDE"
3520 CALL KEY(3,EN,ST)
3530 IF ST=0 THEN 3520
3540 IF EN=13 THEN 1100 ELSE END
```


aa electronic gmbh

Neue Preise # 5

„SIEMENS" der „Leise":	
Siemens PT 88 T-1 (Tintenstrahl), 8bit/parallel	1745,—
APPLE-BUS COMPUTERPLATINEN + PERIPHERIE	
Motherboard durchkontaktiert mit Lötstoplack	89,—
M-board 48k, gesockelt, voll bestückt + geprüft	595,—
Getaktetes Netzteil geprüft + kurzschlußfest	165,—
M-board 48k + Get.-Netzteil (fertig + gepr.)	750,—
M-board 48k + Netzteil + Tastatur (fertig + geprüft)	870,—
M-board 48k + Get.-Netzteil + Tastatur (fertig + gepr.)	875,—
APOLLO II **Kunststoff-Leergehäuse**	145,—
APOLLO II **Tastatur** incl. Groß/Klein-Taste	155,—
16 K-Speicher-Karte Leerplatine	27,50
16K-Speicher-Karte geprüft	137,—
Z-80 CPU-Karte Leerplatine	27,50
Z-80 CPU-Karte geprüft	137,—
Controller DOS 3.3-Karte Leerplatine	27,50
Controller DOS 3.3-Karte geprüft	137,—
Auto-Controller DOS 3.2/3.3-Karte gepr.	198,—
Parallel-Drucker-Karte Leerplatine	27,50
Parallel-Drucker-Karte + Kabel geprüft	137,—
Par.-Drucker-Karte + 64 K-Buffer-Karte + Kabel gepr.	595,—
Seriell-Interface-Karte V 24 Leerplatine	27,50
Seriell-Interface-Karte V 24 geprüft	195,—
80-Zeichen/24-Zeilen-Karte Leerplatine	27,50
80-Zeichen/24-Zeilen-Karte geprüft	175,—
80Z/24Z-Karte + Softswitch-Schalter Leerplatine	69,—
80Z/24Z-Karte + Softswitch-Schalter geprüft	245,—
PAL-Modulator/Color-Karte Leerplatine	27,50
PAL-Modulator/Color-Karte geprüft	175,—
128 K-Speicher-Leerkarte + Software	198,—
128 K-Speicher-Karte + Software geprüft	695,—
256K-Speicher-Leerkarte + 4 Spezial ICs + Software	375,—
256 K-Speicher-Karte + Software geprüft	1495,—
Eprom-Burner (2708/16/32/64) geprüft	275,—
Entwicklungslochraster-Karte (185 × 70 mm)	69,—
Ab 5 Leerplatinen minus 10 %, ab 20 ./. 20 % Mengenrabatt !!	
Ab 5 geprüfte Platinen minus 5 % Mengenrabatt !!!	
UHF/TV-Modulator universell	27,50
Lüfter anclipsbar (220V)	87,—
APOLLO II (48 K) + UHF-Mod. + Gr/Kl	978,—
APOLLO II (64 K) + UHF-Modulator + Gr/Kl	1098,—
APOLLO II (192 K) + UHF-Modulator + Gr/Kl	1698,—
APOLLO II (64 K) + Disk + Contr. + 12" Monitor	2198,—
APOLLO IIA (48 K) + Gr/Kl + 10er Tastatur	1145,—
APOLLO IIA (64 K) + Gr/Kl + 10er Tastatur	1275,—
APOLLO IIA (192 K) + Gr/Kl + 10er Tastatur	1845,—
APOLLO IIA (64 K) + Disk + Contr. + 12" Monitor	2355,—
APOLLO II-FORTH (48 K)	1048,—
APOLLO IIA-FORTH (48 K)	1195,—
Aufpreis f. Funktionstastatur f. APOLLO IIA	50,—
APOLLO II ASKF (48 K) Sep. Keyboard + Gr/Kl deutscher Zeichen-S. + 10er Bl. + Funkt.-Tasten in „IBM-look like" Gehäuse mit Platz für 2 bzw. 4 Laufw.	1695,—
APOLLO II ASKF (64 K) wie oben	1795,—
APPLE II + Euro (48 K) + UHF-Modulator	2199,—
APPLE II + Euro (64 K) + UHF-Modulator	2339,—
APPLE-Disk II Laufwerk + Controller + Kab. DOS 3.3	1095,—
APPLE-Disk II (2. Laufwerk)	895,—
Disk II + Contr. + Kabel DOS 3.3 (Siemens) im Geh.	798,—
Disk II (2. Laufwerk), (Siemens)	655,—
Disk IIF + Contr. + Kabel DOS 3.3 (½ Höhe) im Geh.	719,—
Disk IIF (2. Laufwerk) (½ Höhe) i. Geh.	585,—
APPLESOFT + Tutorial + Reference Hb. engl.	75,—
APPLE-DOS 3.3 Manual engl.	25,—
APPLE-Pascal Reference + Operating Manual engl.	69,—
APPLE-FORTH Manual engl.	69,—
APPLE-Fortran Manual engl.	69,—
CP/M-Software Vol. I + II Manual engl.	69,—
APPLE II Pascal Handbuch deutsch (te-wi)	56,—
APPLE II Anwenderhandbuch deutsch (te-wi)	49,—
6502-Programmier Handbuch deutsch	45,—
6502-Hardware Handbuch deutsch	39,—
6502-Anwendungen Handbuch deutsch (Rod.Zaks)	38,—
APPLE IIe (64 K)	2675,—
APPLE IIe (128 K) + 80Z/24Z-Karte + Softswitch	2965,—
APPLE IIe (64 K) + Disk + Contr. + 12" Monitor	3725,—
80Z/24Z + Softswitch + 64 K-RAM-Karte geprüft	298,—
TEXAS-Software ab 5 Stück 10 % Rabatt-Preisliste anfordern!	
SHARP PC + CE Serie auf Anfrage	
VC 64 (64 K)	759,—
VC 1530 Datassette	149,—
VC 1541 Floppylaufwerk	745,—
VC 20/VC 64-8bit parallel-Drucker-Int. universell	245,—
CBM 8032 SK (bewegliche Tastatur)	2275,—
Colour-Genie EG 2000 (16 K)	575,—
EXIDY SORCERER (48 K)	1795,—
Seikosha GP-100, 8bit/parallel	695,—
Seikosha GP-250, 8bit/parallel	895,—
Seikosha GP-700A, 8bit/parallel, 7 Farben	1375,—
CP 80 EPSON-komp. Nadel-Drucker m. Carbonb.	949,—
STAR Gemini, 8bit/parallel	1075,—
EPSON RX 80, 8bit/parallel	939,—
EPSON RX 80F/T, 8bit/parallel	1195,—
EPSON FX 80, 8bit/parallel	1535,—
EPSON-Traktoraufsatz für FX 80	129,50
APPLE/EPSON-Drucker Interface + Kabel	137,—
APPLE/EPSON-Drucker Graphik-Interface + Kabel	295,—
CBM/EPSON Interface + Kabel	350,—
MX 80/82, FX/RX 80 Spezialfarbband-Kassette	15,95
Olivetti Praxis 35 + Centronics Interface	1345,—
BROTHER-Typenraddrucker HR 15 8bit/parallel	1650,—
TF 50-Formulartraktor für HR 15	375,—
CF 50-Einzelblattzuführung für HR 15	998,—
KB 50-Tastatur für HR 15	475,—
BMC 12" Video-Monitor gr. 18 MHz	285,—
SANYO DM 2112 12" grün 15 MHz	285,—
SANYO DM 2212 12" orange 15 MHz	395,—
SANYO DD 8112 12" grün 18 MHz	475,—
SANYO DD 8212 12" orange 18 MHz	485,—
SANYO CM 3185 12" Farbm. + RGB Ind.-Qual.	935,—
APPLE II Video-Mon. 12", 20 MHz, entspiegelt	499,—
3M-Scotch Disketten-Verstärkt in Box I. Wahl	
5¼" 1D-Disk-Soft-o.Hardsektoriert 10er Pack	65,—
5¼" 1D-Disk-Soft-o.Hardsektoriert 100er Pack	629,—
5¼" 2D-Disk-Softsektoriert 10er Pack	97,—
5¼" 2D-Disk-Softsektoriert 100er Pack	939,—
5¼" 1/96 Softsektoriert 10er Pack	91,95
5¼" 1/96 Softsektoriert 100er Pack	889,—
5¼" 2/96 Softsektoriert 10er Pack	123,95
5¼" 2/96 Softsektoriert 100er Pack	1199,—
8" Soft-o. Hardsektoriert 10er Pack	67,—
8" Soft-o. Hardsektoriert 100er Pack	649,—
8" 1D-Disk-Softsektoriert 10er Pack	86,—
8" 1D-Disk-Softsektoriert 100er Pack	829,—
8" 2D-Disk-Softsektoriert 10er Pack	99,—
8" 2D-Disk-Softsektoriert 100er Pack	959,—
3M-Scotch 5¼" Reinigungsset (2 Disketten)	67,95
BASF Disketten in Box + Aufkleber I. Wahl	
Aufschlag für Verstärkungsring: –,30 DM pro Diskette	
5¼" o.8" Soft-o.Hardsektoriert 10er Pack	48,—
5¼" o.8" Soft-o.Hardsektoriert 100er Pack	469,—
5¼" o.8" 1D-Softsektoriert 10er Pack	54,—
5¼" o.8" 1D-Softsektoriert 100er Pack	529,—
5¼" o.8" 2D-Softsektoriert 10er Pack	74,—
5¼" o.8" 2D-Softsektoriert 100er Pack	719,—
5¼" 1/96 Softsektoriert 10er Pack	69,—
5¼" 1/96 Softsektoriert 100er Pack	679,—
5¼" 2/96 Softsektoriert 10er Pack	85,—
5¼" 2/96 Softsektoriert 100er Pack	829,—
Reinigungs-Diskette	59,95
5¼" Disketten-Archivbox für 10Disks	7,50
5¼" Disketten-Archivbox 10er Pack	72,—
8" Disketten-Archivbox für 10 Disks	9,50
8" Disketten-Archivbox 10er Pack	89,—
5¼" Disk-Karteikasten Kunststoff (ca. 80 Disketten)	34,95
5¼" Disk-Karteikasten Rauchglas (ca. 80 Disketten)	49,95
5¼" Disk-Karteik. Rauchglas, abschließbar (ca. 80 D.)	67,95
8" Disk-Karteikasten Rauchglas (ca. 80 D.)	74,95
2000 Bl. Tabelierpapier (240 × 12" einfach) weiß oder grün/weiß perforiert	47,—
4000 Etiketten-aufkleber (107 × 36 mm Doppelreihe) auf 240 mm × 12" perf. Trägerpapier	66,50
16000 Etiketten-Aufkleber w.o.	249,—
10000 Etiketten-Aufkleber m.sh. (125 × 12")	179,—
EPROM-Löschlampe E-27 universell	79,95
OSZILLOSCOPE HAMEG ab Lager (s.V.r.)	

Bei Vorauszahlung frei Empfangsstation unversichert in der BRD, ausgenommen Papier und Etiketten, sonst N.N. + V.S.

aa electronic gmbh

Telex: 07 72 642 aaa-d
Habsburger Straße 134
7800 FREIBURG, Tel. (07 61) 27 68 64
Bauelemente – Bausätze – µP's
Meßgeräte – Zubehör – Fachliteratur
Fachgeschäft für Elektronik + Mikrocomputer

Das Transportproblem gelöst

TI 99/4A

Das nachfolgende Programm kommt aus dem Bereich des Operations Research und ist ein Beispiel für die Lösung von Optimierungsproblemen. Konkret geht es hier um die Minimierung von Transportkosten.

Das Programm »Transportproblem« zeigt deutlich die Grenzen einer interpretierenden Programmiersprache, wie es Basic auch ist. Um die für das Programm benötigte Datenmenge zu verarbeiten, werden einige Minuten benötigt (eine compilierte Programmiersprache wie beispielsweise Fortran, würde für ein und die selbe Aufgabe nur Sekunden benötigen). Umso weniger Daten also bearbeitet werden, umso schneller wird die Lösung gefunden.

Meine Absicht ist nicht, den Algorithmus bis ins Detail zu erklären, sondern vielmehr das Problem zu erläutern, den mathematischen Ansatz herzuleiten und die Anwendung des Programms zu beschreiben.

Das Problem und sein Lösungsansatz

Das Transportproblem läßt sich folgendermaßen formulieren: An n Orten sind die Mengen b(j) einer Ware vorrätig und an m weiteren Orten werden von dieser Ware die Mengen a(i) benötigt. Der Gesamtbedarf sei gleich dem Gesamtvorrat, also

$$\sum_{i=1}^{m} a(i) = \sum_{j=1}^{n} b(j)$$

Die Transportkosten pro Mengeneinheit vom Lagerort j zum Bedarfsort i seien c(i,j), die von j nach i zu transportierenden Einzelmengen seien z(i,j). Es müssen also die Einzelkosten bestimmt werden, für die die Gesamttransportkosten minimalsind.

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c(i,j) \times z(i,j)$$

Zwei weitere Bedingungen ergeben sich aus der Forderung, daß die Summe der zum Ort i hin- und vom Lager j wegzutransportierenden Einzelmengen gleich den benötigten, beziehungsweise den verfügbaren Mengen seien, also

$$\sum_{i=1}^{m} z(i,j) = a(i) \text{ und}$$

$$\sum_{j=1}^{n} z(i,j) = b(j)$$

mit i=1....m und j=1....n

Die Variable w enthält den Wert der zu minimierenden Zielfunktion. Der vorliegende Algorithmus beruht übrigens auf einer von Hadley beschriebenen Lösungsmethode.

Listing zum »Transportproblem« ►

```
100 !**********************
110 !*  TRANSPORTPROBLEM   *
120 !*                     *
130 !*  @ DANIEL RUESTER   *
140 !*                     *
150 !*  JANUAR 1984        *
160 !**********************
170 !Extended Basic
180 OPTION BASE1
190 ON WARNING NEXT :: ON BREAK
    NEXT :: ON ERROR 360
200 GOTO 260 :: H,I,J,K,L,M,N,S,
    V,W,X,Y,SA,SB,CH1,CH2=CH3 ::
     D$=M$
210 CALL CHAR :: CALL HCHAR :: C
    ALL VCHAR :: CALL CLEAR :: C
    ALL COLOR :: CALL SCREEN ::
    CALL KEY :: CALL SOUND
220 CALL SPRITE :: CALL JOYST ::
     CALL POSITION :: CALL MOTIO
    N :: CALL DELSPRITE :: CALL
    TRANSP :: CALL RAHMEN
230 DIM C(12,5),CC(12,5),C$(12,5
    ),A(12),AA(12),B(5),BB(5),Z(
    12,5),A$(13),FR(3)
240 DATA - EINGEBEN,- EINLESEN,-
     KORRIGIEREN,- SPEICHERN,VOR
    RAT/BEDARF,ALGORITHMUS START
    EN,ENDE,KOSTENMATRIX
250 !@P-
260 DATA - LAGER,- FILIALE,'W' F
    UER WAHLLISTE,ANZAHL DER,* D
    ATENKASSETTE EINLEGEN!
270 DATA 524,392,262
280 CALL CHAR(92,"003844BAA2BA44
    38")
290 !titel
300 CALL RAHMEN(2,11,6):: DISPLA
    Y AT(7,3):"EIN OPERATIONS-RE
    SEARCH-" :: DISPLAY AT(10,9)
    :"ALGORITHMUS:"
310 DISPLAY AT(16,5):"DAS TRANSP
    ORTPROBLEM" :: DISPLAY AT(21
    ,7):"\ DANIEL RUESTER"
320 FOR I=1 TO 13 :: READ A$(I):
    : NEXT I :: FOR J=1 TO 3 ::
    READ FR(J):: NEXT J
330 CALL CHAR(128,"000000FFFF000
    000181818181818FF8181818
    18181FF")
```

Listing zum »Transportproblem« (Fortsetzung)

```
340 FOR I=1 TO 1000 :: NEXT I
350 !menue
360 CALL DELSPRITE(#1):: CALL RA
    HMEN(2,12,8)
370 DISPLAY AT(5,2)BEEP:A$(8)
380 FOR I=1 TO 7 :: DISPLAY AT(5
    +2*I,2):A$(I);TAB(24);I :: N
    EXT I
390 CALL KEY(3,K,S):: IF S=0 OR
    K<49 OR K>55 THEN 390
400 ON K-48 GOSUB 450,550,700,80
    0,990,1170,1340
410 DISPLAY AT(24,1):A$(11)
420 CALL KEY(3,K,S):: IF S=0 OR
    K<>87 THEN 420 ELSE 360
430 STOP
440 !kostenm. eingeben
450 CALL RAHMEN(14,16,3)
460 DISPLAY AT(1,1):A$(8),A$(1):
    : DISPLAY AT(9,1):A$(12): :
    :A$(9): : :A$(10)&"N"
470 ACCEPT AT(12,18)BEEP(DIGIT):
    N :: IF N>5 OR N<2 THEN 470
480 ACCEPT AT(15,18)BEEP(DIGIT):
    M :: IF M>12 OR M<2 THEN 480
490 DISPLAY AT(9,1):"": : :"": :
     :"" :: GOSUB 1350
500 FOR I=1 TO M :: FOR J=1 TO N
510 ACCEPT AT(5+I,J*4)(DIGIT)SIZ
    E(3):C(I,J)
520 NEXT J :: NEXT I
530 CH1=1 :: RETURN
540 !kostenm. einlesen
550 CALL RAHMEN(16,5,3)
560 PRINT A$(8),A$(2): :"* DATEI
    -LESEKONTROLLE *": :"* DATEN
    KASSETTE EINLEGEN!": :"  DAT
    EI EINLESEN (J/N)?  _"
570 CALL KEY(3,K,S):: IF S=0 THE
    N 570 ELSE IF K=78 THEN 360
    ELSE IF K<>74 THEN 570 ELSE
    DISPLAY AT(23,26):"J"
580 OPEN #1:"CS1",INTERNAL ,INPU
    T ,FIXED 192
590 INPUT #1:D$
600 CLOSE #1
610 N=VAL(SEG$(D$,1,1)):: M$=SEG
    $(D$,2,POS(D$,"#",1)-2):: M=
    VAL(M$)
620 L=LEN(M$)+3
630 FOR I=1 TO M :: FOR J=1 TO N
640 C(I,J)=VAL(SEG$(D$,L,3))
650 L=L+3 :: NEXT J :: NEXT I
660 PRINT : :" DATENBLOCK:":D$:
    :" EINZELDATEN:"
670 FOR I=1 TO M :: FOR J=1 TO N
     :: PRINT C(I,J);:: NEXT J :
```

Anwendung des Programms

Nach einem kurzzeitigen Titelbild erscheint das Menü. Es bietet vier Möglichkeiten an, mit der Kostenmatrix zu arbeiten. Man kann die Bedarfsgrößen der Filialen und die Vorratsmengen der einzelnen Lager eingeben, sowie den Rechenvorgang starten und das Programm beenden. Kontrollvariablen sorgen für einen geordneten Programmablauf, so daß es zum Beispiel nicht möglich ist, die Funktionen 3 bis 6 durchzuführen, bevor die Option 1 oder 2 angewählt wurde.

Option 1: Kostenmatrix eingeben

Das Programm fragt zunächst nach der Anzahl der (lieferfähigen) Lager. Es werden nur Werte zwischen 2 und 5 angenommen, denn hätte man nur ein Lager zur Verfügung, würde die Lösung auf der Hand liegen, das heißt, die zur Verfügung stehenden Einzelmengen wären sofort ersichtlich und bräuchten nicht mit einem Computer berechnet zu werden. Die Eingabe der Anzahl der Lager ist mit der »ENTER« Taste abzuschließen. Dann fragt das Programm nach der Anzahl der zu beliefernden Filialen. Diese darf zwischen 2 und 12 liegen.

Die Beschränkung auf 5 Lager erfolgte übrigens wegen des begrenzenden Platzes auf dem Bildschirm des TI 99/4A, die der Filialen, aus Speicherplatzgründen. Hierdurch wird die Realitätsbezogenheit nicht verletzt, denn für kleine bis mittlere regionale Filialunternehmen wäre das Programm »Transportproblem« durchaus anwendbar.

Sind die beiden Werte eingegeben, wird auf eine Tabelle umgeschaltet, in die die Kosten-Matrix-Elemente eingegeben werden. Die einzelnen Lager werden mit den Zahlen 1 bis 5 und die Filialen mit den Buchstaben A bis L gekennzeichnet. Der blinkende Cursor fordert zur Eingabe der Werte auf. Die Abfrage beginnt mit Element A1, das die Transportkosten pro Mengeneinheit von Lager 1 zur Filiale A angibt. Die Eingabe erfolgt zeilenweise. Der Cursor wird automatisch, nach Betätigen der »ENTER« Taste, zur nächsten Eingabeposition bewegt. Es können nur ganze Zahlen zwischen 0 und 999 eingegeben werden. Daß diese Beschränkung die Anwendbarkeit des Programms nicht begrenzt, zeigt das Beispiel eines Programmablaufs. Sind alle Daten eingegeben, gelangt man nach Drücken von W zurück zur Wahlliste.

Option 2: Kostenmatrix einlesen

Eine Dateikontrolle führt durch diesen Programmabschnitt. Außerdem wird man aufgefordert eine Datenkassette in den Kassettenrecorder einzulegen. Steht keine Kostenmatrix auf Band zur Verfügung, kann man durch Drücken der Taste N zum Menü zurück gelangen. Drückt man J für ja, erscheint die bekannte Aufforderung vom Betriebssystem des TI 99/4A, nämlich die Kassette einzulegen und die »PLAY« Taste zu betätigen. Ist dann keine Datenkassette eingelegt, oder sollten keine Daten auf Kassette zur Verfügung stehen, wird der Bildschirmaufbau zerstört. Eine Möglichkeit dies zu unterdrücken, habe ich noch nicht gefunden.

Sollten aber Daten auf Band zur Verfügung stehen, werden diese auf dem Bildschirm ausgedruckt. Man beachte: Die erste Ziffer steht für die Anzahl der Lager, die darauffolgende Zahl, bis zum Doppelkreuz, für die Anzahl der Filialen. (Siehe hierzu auch Programmteil 4: Speichern der Matrix). Außerdem kann man noch überprüfen, ob die Daten korrekt eingelesen und korrekt aufgeteilt wurden. Mit Druck auf die Taste W gelangt man wieder zum Menü zurück.

Listing zum »Transportproblem« (Fortsetzung)

```
    : NEXT I
680 CH2=1 :: PRINT : : ::: RETUR
    N
690 !kostenm. korrigieren
700 CALL RAHMEN(5,16,9):: IF CH1
    =0 AND CH2=0 THEN GOSUB 1180
     :: RETURN
710 DISPLAY AT(1,1):A$(8),A$(3):
    : GOSUB 1350
720 FOR I=1 TO M :: FOR J=1 TO N
     :: DISPLAY AT(I+5,J*4-1):C(
    I,J):: NEXT J :: NEXT I
730 CALL SPRITE(#1,130,7,41,49)
740 DISPLAY AT(21,2):"CURSORPOSI
    TION:": : :A$(11)
750 CALL JOYST(1,X,Y):: CALL KEY
    (1,K,S):: IF K=5 THEN Y=4 EL
    SE IF K=0 THEN Y=-4 ELSE IF
    K=2 THEN X=-4 ELSE IF K=3 TH
    EN X=4
760 CALL MOTION(#1,-2*Y,8*X):: C
    ALL POSITION(#1,V,H):: V=INT
    ((V+3)/8)-4 :: H=INT(H/32)::
     IF V>M OR V<1 OR H>N OR H<1
     THEN V=-16 :: H=0
770 DISPLAY AT(21,18):CHR$(V+64)
    ;H :: IF K=4 THEN RETURN ELS
    E IF H=0 OR K<>18 THEN 750
780 ACCEPT AT(V+5,H*4)(DIGIT)BEE
    PSIZE(3):C(V,H):: GOTO 750
790 !kostenm. speichern
800 CALL RAHMEN(16,6,10):: IF CH
    1=0 AND CH2=0 THEN GOSUB 118
    0 :: RETURN
810 PRINT A$(8),A$(4): :"* DATEI
    -SPEICHERKONTROLLE": :A$(13)
820 D$=STR$(N)&STR$(M)&"#"
830 FOR I=1 TO M :: FOR J=1 TO N
     :: C$(I,J)=STR$(C(I,J))
840 ON LEN(C$(I,J))GOSUB 860,870
    ,880
850 NEXT J :: NEXT I :: GOTO 890
860 C$(I,J)="00"&C$(I,J):: RETUR
    N
870 C$(I,J)="0"&C$(I,J):: RETURN
880 RETURN
890 PRINT : :" EINZELDATEN:"
900 FOR I=1 TO M :: FOR J=1 TO N
     :: PRINT C(I,J);:: NEXT J :
    : NEXT I
910 FOR I=1 TO M :: FOR J=1 TO N
     :: D$=D$&C$(I,J):: NEXT J :
    : NEXT I
920 PRINT : :" DATENBLOCK:":D$:
    :"  DATEI SPEICHERN (J/N)? _
    "
930 CALL KEY(3,K,S):: IF S=0 THE
    N 930 ELSE IF K=78 THEN 360
940 IF K<>74 THEN 930 ELSE DISPL
    AY AT(23,26):"J"
950 OPEN #2:"CS1",INTERNAL ,OUTP
    UT ,FIXED 192
960 PRINT #2:D$
970 CLOSE #2 :: GOTO 360
980 !vorrat/bedarf eingeben
990 CALL RAHMEN(14,16,13):: IF C
    H1=0 AND CH2=0 THEN GOSUB 11
    80 :: RETURN
1000 SA,SB=0 :: DISPLAY AT(1,1):
     A$(5),A$(1)
1010 DISPLAY AT(3,1):"VORRAT"
1020 FOR J=1 TO N :: DISPLAY AT(
     2+J,8):A$(9);J :: NEXT J
1030 DISPLAY AT(4+N,1):"BEDARF"
1040 FOR I=1 TO M :: DISPLAY AT(
     3+N+I,8):A$(10);" ";CHR$(I+
     64):: NEXT I
1050 FOR J=1 TO N :: ACCEPT AT(2
     +J,23)(DIGIT)SIZE(3):B(J)
1060 SB=SB+B(J):: DISPLAY AT(4,2
     )SIZE(5):SB :: NEXT J
1070 FOR I=1 TO M :: ACCEPT AT(3
     +N+I,23)(DIGIT)SIZE(3):A(I)
1080 SA=SA+A(I):: DISPLAY AT(5+N
     ,2)SIZE(5):SA :: NEXT I
1090 DISPLAY AT(24,2):"KORREKTUR
      (A - ";CHR$(M+64);"/N)?"
1100 CALL KEY(3,K,S):: IF K=78 T
     HEN DISPLAY AT(24,24):"N" :
     : GOTO 1150
1110 IF K<65 OR K>M+64 THEN 1100
1120 DISPLAY AT(24,24):CHR$(K)::
      K=K-64 :: SA=SA-A(K):: DIS
     PLAY AT(5+N,2)SIZE(5):SA
1130 ACCEPT AT(3+N+K,23)(DIGIT)S
     IZE(3):A(K)
1140 SA=SA+A(K):: DISPLAY AT(5+N
     ,2)SIZE(5):SA :: GOTO 1100
1150 IF SA<>SB THEN CALL SOUND(2
     00,440,0):: GOTO 1100 ELSE
     CH3=1 :: GOTO 360
1160 !algorithmus starten
1170 CALL RAHMEN(2,9,3):: IF CH1
      OR CH2 THEN 1190
1180 DISPLAY AT(10,1)BEEP:" ACHT
     UNG!": :" ERST ";A$(8): :"
     EINGEBEN ODER EINLESEN!" ::
      RETURN
1190 IF CH3=0 THEN 990
1200 DISPLAY AT(1,1):"ALGORITHMU
     S GESTARTET!" :: GOSUB 1350
1210 FOR I=1 TO M :: FOR J=1 TO
     N :: DISPLAY AT(5+I,4*J-1):
     C(I,J):: NEXT J :: DISPLAY
     AT(5+I,24):A(I):: NEXT I
```

```
1220 FOR J=1 TO N :: DISPLAY AT(
     20,4*J-1):B(J):: NEXT J
1230 DISPLAY AT(20,24):SB
1240 FOR I=1 TO M :: FOR J=1 TO
     N :: CC(I,J)=C(I,J):: NEXT
     J :: AA(I)=A(I):: NEXT I
1250 FOR J=1 TO N :: BB(J)=B(J):
     : NEXT J
1260 CALL TRANSP(M,N,A(),B(),C(,
     ),Z(,),W)
1270 FOR I=1 TO 3 :: FOR J=1 TO
     3 :: FOR K=0 TO 20 STEP 4 :
     : CALL SOUND(-100,FR(J),K):
     : NEXT K :: NEXT J :: NEXT
     I
1280 DISPLAY AT(1,1):"LOESUNG:"
```

Option 3: Kostenmatrix korrigieren

Um das Programm komfortabel zu gestalten, darf eine Korrekturmöglichkeit einzelner Daten, ohne alle Daten erneut eingeben zu müssen, natürlich nicht fehlen. Eine Korrektur ist zum einen bei einer fehlerhaften Eingabe nötig, und zum anderen, wenn sich in einer relativ konstanten Kostenmatrix nur wenige Elemente verändern.

Zum Korrigieren wird die Matrix, so wie sie eingegeben wurde, auf dem Bildschirm dargestellt. Zuerst erscheint auf Position A 1 ein kleines Kästchen, das die jeweilige zu korrigierende Stelle angibt. Dieses Kästchen kann wie folgt bewegt werden:
1) mit den Cursortasten
2) mit Joystick 1

Mit den Pfeiltasten wird eine Bewegung in vier Richtungen ermöglicht, mit dem Joystick kann der Cursor auch diagonal bewegt werden. Das Kästchen ist übrigens ein Sprite. Die Geschwindigkeit des Sprites ist so bemessen, daß man innerhalb kurzer Zeit jede beliebige Stelle auf dem Bildschirm erreichen kann, ohne über den Rand hinauszuschießen. Mit ein wenig Übung wird man sicherlich schnell die nötigen Korrekturen vornehmen können. Das Sprite kann über den ganzen Bildschirm bewegt werden, Änderungen können aber nur innerhalb der Tabelle vorgenommen werden. Um das Arbeiten zu erleichtern, wird stets die aktuelle Cursorposition angezeigt. Befindet sich das Sprite außerhalb des Korrekturbereichs, wird als Position 0 0 angezeigt. Eine Besonderheit des Programms »Transportproblem« besteht darin, daß man das Sprite nicht exakt mit dem zu korrigierenden Zeichen in Deckung bringen muß. Zur Kontrolle kann man immer die angegebene Cursorposition zur Hilfe nehmen. Hat man die zu korrigierende Stelle erreicht, betätigt man die Taste Q, oder beim Joystick die Feuertaste. Danach erscheint der blinkende Cursor und fordert zur Eingabe der richtigen Zahl (0 bis 999) auf. Nach dem Drücken der »ENTER« Taste verschwindet der Cursor und die nächste Korrekturposition kann angesteuert werden. Nach Betätigung der W Taste gelangt man wieder ins Hauptmenü zurück. Durch nochmaliges Anwählen der Korrekturroutine kann man nochmals seine Verbesserungen betrachten.

Option 4: Kostenmatrix speichern

Es ist nur dann möglich eine Kostenmatrix abzuspeichern, wenn diese vorher von Band eingelesen oder manuell eingegeben wurden. Der Speichervorgang verläuft entsprechend dem Lesevorgang. Zuerst werden die Einzeldaten auf den Bildschirm gebracht, anschließend der Datenblock, dem die Anzahl der Lager und der Filialen beigefügt sind. Auch hier wird, falls versehentlich die Taste 4 betätigt wurde, die Möglichkeit gegeben ins Menü zurück zu gelangen. Wird die Taste J betätigt, folgt der bekannte Speichervorgang und dann der Rücksprung ins Menü.

Option 5: Vorrat/Bedarf

Die einzelnen Lager und Filialen werden aufgelistet und dann auf eine Eingabe gewartet. Hinter den angegebenen Lagern gibt man die vorhandenen Lagerreserven ein, hinter den Filialen die benötigten Mengen der Ware. Es werden nur ganze Zahlen zwischen 0 und 999 akzeptiert. Mit jeder Eingabe wird die Zwischensumme des Vorrats unter »Vorrat«, die der Bedarfsmengen, unter »Bedarf« angegeben. Eine Bedingung des Programms »Transportproblem« ist, wie schon erwähnt, daß die Gesamtvorratsmenge gleich der Gesamtbedarfsmenge sein muß. Sind nach beendeter Eingabe beide Summen nicht gleich, so muß korrigiert werden. Man wählt die Filiale aus, deren Menge geändert werden soll und drückt zur Korrektur die entsprechende Taste. Der Cursor erscheint und fordert zur Eingabe auf. Vorher wird von der Summe des Gesamtbedarfs der zu korrigierende Wert abgezogen, so daß man die Summen vergleichen kann und entsprechende Werte eingeben kann. Nach der Eingabe wird der neue Wert sofort wieder addiert. Nach Drücken von »N« für Nein, wird der Programmteil verlassen, aber nur, wenn die beiden Summen gleich sind.

Easifeed 230

die Einzelblattzuführung
für alle low-cost Printer

Bezugsnachweis:
Multimatic AG
CH-8700 Küsnacht-Zürich

multi matic

Listing zum »Transportproblem« (Fortsetzung)

```
1290 FOR I=1 TO M :: FOR J=1 TO
     N :: DISPLAY AT(5+I,4*J-1)S
     IZE(4):Z(I,J):: NEXT J :: N
     EXT I
1300 DISPLAY AT(22,2):"GESAMTKOS
     TEN: ";W
1310 FOR I=1 TO M :: FOR J=1 TO
     N :: C(I,J)=CC(I,J):: NEXT
     J :: A(I)=AA(I):: NEXT I
1320 FOR J=1 TO N :: B(J)=BB(J):
     : NEXT J
1330 CH3=0 :: RETURN
1340 CALL CLEAR :: STOP
1350 CALL COLOR(13,2,1):: CALL H
     CHAR(4,3,128,23):: CALL VCH
     AR(3,4,129,16)
1360 FOR I=1 TO M :: CALL VCHAR(
     5+I,3,I+64):: NEXT I
1370 FOR I=1 TO N :: CALL HCHAR(
     3,3+I*4,I+48):: NEXT I
1380 RETURN
1390 !@P+
1400 SUB RAHMEN(ZF,BF,RF):: CALL
      CLEAR :: CALL VCHAR(1,1,14
     3,48):: CALL VCHAR(1,31,143
     ,48):: CALL COLOR(14,RF,RF)
     :: CALL SCREEN(RF)
1410 FOR I=1 TO 8 :: CALL COLOR(
     I,ZF,BF):: NEXT I :: SUBEND
1420 !transportproblem
1430 SUB TRANSP(M,N,A(),B(),C(,)
     ,Z(,),W)
1440 GOTO 1470 :: G,H,I,J,K,L,AI
     ,BJ,IP,IS,IT,IU,IV,IZ,JT,N1
     ,IN1,INF,LW1,LWJ,LWN,NVI,ZI
     J=CIJ
1450 DIM IG(12),LU(12),NV(12),IR
     (5),LV(5),LS(61),NL(60),LW(
     6)
1460 !@P-
1470 INF=99999999 :: N1=N+1
1480 FOR I=1 TO M :: NV(I)=(I-1)
     *N :: FOR J=1 TO N :: Z(I,J
     )=0 :: NEXT J :: NEXT I
1490 FOR J=1 TO N :: LV(J)=1 ::
     LW(J)=0 :: NEXT J :: LW(N1)
     =0
1500 W,G=0 :: FOR I=1 TO M :: H=
     INF :: FOR J=1 TO N :: IF C
     (I,J)<H THEN H=C(I,J)
1510 NEXT J
1520 FOR J=1 TO N :: C(I,J)=C(I,
     J)-H :: IF C(I,J)THEN 1540
1530 LV(J)=0 :: NV(I)=NV(I)+1 ::
      NVI=NV(I):: NL(NVI)=J
1540 NEXT J :: W=H*A(I)+W :: NEX
     T I
1550 FOR J=1 TO N :: IF LV(J)=0
     THEN 1610
1560 H=INF :: FOR I=1 TO M :: IF
      C(I,J)<H THEN H=C(I,J)
1570 NEXT I
1580 FOR I=1 TO M :: C(I,J)=C(I,
     J)-H :: IF C(I,J)THEN 1600
1590 NV(I)=NV(I)+1 :: NVI=NV(I):
     : NL(NVI)=J
1600 NEXT I :: W=H*B(J)+W
1610 NEXT J :: FOR I=1 TO M :: A
     I=A(I):: NVI=NV(I):: IN1=(I
     -1)*N+1 :: IF IN1>NVI THEN
     1690
1620 FOR IU=IN1 TO NVI :: IF AI=
     0 THEN 1690
1630 J=NL(IU):: BJ=B(J):: IF BJ=
     0 THEN 1680
1640 H=BJ :: IF AI<BJ THEN H=AI
1650 Z(I,J)=H :: AI=AI-H :: B(J)
     =BJ-H :: LWJ=LW(J+1):: LWN=
     LW(N1):: IF LWJ>LWN THEN 16
     70
1660 FOR IT=LWJ TO LWN :: JT=LWN
     -IT+LWJ+1 :: LS(JT+1)=LS(JT
     ):: NEXT IT
1670 FOR IT=J TO N :: JT=IT+1 ::
      LW(JT)=LW(JT)+1 :: NEXT IT
      :: LS(LWJ+2)=I
1680 NEXT IU
1690 A(I)=AI :: G=G+AI :: NEXT I
1700 IF G=0 THEN 2250
1710 FOR J=1 TO N :: IR(J)=0 ::
     NEXT J :: K=0 :: FOR I=1 TO
      M :: IF A(I)=0 THEN 1730
1720 K=K+1 :: LU(K)=I :: IG(I)=I
     NF :: GOTO 1740
1730 IG(I)=0
1740 NEXT I
1750 L=0 :: IF K=0 THEN 1810
1760 FOR IU=1 TO K :: I=LU(IU)::
      NVI=NV(I):: IN1=(I-1)*N+1
     :: IF IN1>NVI THEN 1800
1770 FOR IS=IN1 TO NVI :: J=NL(I
     S):: IF IR(J)THEN 1790
1780 IR(J)=I :: L=L+1 :: LV(L)=J
      :: IF B(J)>0 THEN 1880
1790 NEXT IS
1800 NEXT IU
1810 IF L=0 THEN 2070
1820 K=0 :: FOR IV=1 TO L :: J=L
     V(IV):: LWJ=LW(J+1):: LW1=L
     W(J)+1 :: IF LW1>LWJ THEN 1
     860
1830 FOR IS=LW1 TO LWJ :: I=LS(I
     S+1):: IF IG(I)THEN 1850
1840 IG(I)=J :: K=K+1 :: LU(K)=I
1850 NEXT IS
1860 NEXT IV :: IF K=0 THEN 2070
```

Listing zum »Transportproblem« (Schluß)

```
1870 GOTO 1750
1880 H=B(J):: IP=J
1890 I=IR(J):: J=IG(I):: IF J<>I
     NF THEN 1920
1900 IF A(I)<H THEN H=A(I)
1910 GOTO 1940
1920 IF Z(I,J)<H THEN H=Z(I,J)
1930 GOTO 1890
1940 J=IP :: B(J)=B(J)-H :: A(I)
     =A(I)-H :: G=G-H
1950 I=IR(J):: ZIJ=Z(I,J):: Z(I,
     J)=ZIJ+H :: IF ZIJ THEN 199
     0
1960 LWJ=LW(J+1):: LWN=LW(N1)::
     IF LWJ>LWN THEN 1980
1970 FOR IT=LWJ TO LWN :: JT=LWN
     -IT+LWJ+1 :: LS(JT+1)=LS(JT
     ):: NEXT IT
1980 FOR IT=J TO N :: JT=IT+1 ::
      LW(JT)=LW(JT)+1 :: NEXT IT
      :: LS(LWJ+2)=I
1990 J=IG(I):: IF J=INF THEN 170
     0
2000 Z(I,J)=Z(I,J)-H :: IF Z(I,J
     )THEN 1950
2010 LWJ=LW(J+1):: LW1=LW(J)+1 :
     : IF LW1>LWJ THEN 2050
2020 FOR IT=LW1 TO LWJ :: IF LS(
     IT+1)<>I THEN 2040
2030 IS=IT :: GOTO 2050
2040 NEXT IT
2050 FOR IT=J TO N :: JT=IT+1 ::
      LW(JT)=LW(JT)-1 :: NEXT IT
      :: LWJ=LW(N1):: IF IS>LWJ
     THEN 1950
2060 FOR IT=IS TO LWJ :: JT=IT+1
      :: LS(JT)=LS(JT+1):: NEXT
     IT :: GOTO 1950
2070 K=0 :: L=N1 :: FOR J=1 TO N
      :: IF IR(J)THEN 2090
2080 K=K+1 :: LV(K)=J :: GOTO 21
     00
2090 L=L-1 :: LV(L)=J
2100 NEXT J :: H=INF :: FOR I=1
     TO M :: IF IG(I)=0 OR K=0 T
     HEN 2130
2110 FOR IS=1 TO K :: J=LV(IS)::
      IF C(I,J)<H THEN H=C(I,J)
2120 NEXT IS
2130 NEXT I :: FOR I=1 TO M :: I
     Z=0 :: IF IG(I)THEN IZ=1
2140 NVI=(I-1)*N :: FOR IS=L TO
     N :: J=LV(IS):: IF IZ<>1 TH
     EN C(I,J)=C(I,J)+H
2150 CIJ=C(I,J):: IF CIJ THEN 21
     70
2160 NVI=NVI+1 :: NL(NVI)=J
2170 NEXT IS :: IF K=0 THEN 2220
2180 FOR IS=1 TO K :: J=LV(IS)::
      IF IZ=1 THEN C(I,J)=C(I,J)
     -H
2190 CIJ=C(I,J):: IF CIJ THEN 22
     10
2200 NVI=NVI+1 :: NL(NVI)=J
2210 NEXT IS
2220 NV(I)=NVI :: NEXT I
2230 W=H*G+W :: GOTO 1710
2240 !@P+
2250 SUBEND
```

Option 6: Algorithmus starten

Hier beginnt der eigentliche Rechenvorgang. Zunächst wird die Kostenmatrix wieder in Tabellenform dargestellt, dann werden die Bedarfsmengen in der letzten Spalte dargestellt und die Vorratsmengen in der letzten Zeile der Matrix. Die rechts unten stehende Zahl gibt die zu transportierende Gesamtmenge an. Solange der Rechenvorgang stattfindet, steht in der Kopfzeile: »Algorithmus gestartet«. Ist die Optimallösung gefunden, ertönt ein Dreifachgong. In der Kopfzeile steht dann »Lösung«. Die Kostenmatrix-Elemente werden durch die gesuchten Einzelmengen ersetzt und die Gesamtkosten ausgedruckt. Die Summe der Elemente einer Zeile muß nun gleich der rechts von der Tabelle angegebenen Zahl sein. Addiert man die Elemente spaltenweise, so ergeben sich daraus die in der untersten Zeile angegebenen Werte. Wie diese Werte zu interpretieren sind, soll das nachfolgende Beispiel veranschaulichen.

Option 7: Beendet das Programm

Bemerkungen:
Sämtliche Zahleneingaben müssen mit »ENTER« abgeschlossen werden. Das Programm »Transportproblem« ist so abgesichert, daß weder ein unlogischer Ablauf, noch eine falsche Eingabe von Daten möglich ist. Auch für den Fall, daß beim Einlesen einer Datei ein Fehler auftritt, es wird dann zur Wahlliste verzweigt. Bei allen numerischen oder alphanumerischen Eingaben , die einen Sprung zu einem bestimmten Programmteil veranlassen, muß nur die entsprechende Taste gedrückt werden (ohne »ENTER«). Mit »BREAK« kann das Programm nicht unterbrochen werden, dies ist nur über die Wahlliste möglich (Option 7).

KAYPRO II
Der Portable Computer des Jahres 1983
mit Drucker
CP80-Matrix-Drucker (Epson-komp.) DM 5995,-
Computer, Drucker, Druckerkabel und Software inkl. MwSt.
inkl. CP/M, Wordstar, Supercalc, MBasic, dBASE II, The Word plus, Uniform, Term
Fa. HMS
T. 089/295991 · Isartorplatz 4 · 8000 München 2

Nun ein Beispiel: Ein Einzelhandelsunternehmen besitzt in zwölf Städten Filialen, in fünf Städten Warenlager.

Die Filialen befinden sich in	Benötigte Mengen
A Bremen	40
B Essen	90
C Hamburg	60
D Karlsruhe	6
E Kiel	17
F Köln	71
G Mannheim	35
H München	83
I Nürnberg	66
J Saarbrücken	13
K Würzburg	8
L Bielefeld	100
Die Lager befinden sich in	**Vorrätige Mengen**
1 Frankfurt	177
2 Hamburg	133
3 Hannover	134
4 Nürnberg	86
5 Stuttgart	59

Beispielsdaten zur Berechnung von Transportkosten.

Die Transportkosten seien proportional zur Entfernung in Kilometern, der Proportionalitätsfaktor sei der Einfachheit halber gleich 1.
Man geht nun so vor:

Anwahl des Programmteils »Kostenmatrix eingeben«. Bei Lager »5« und bei Filialen »12« eingeben. In die Kostenmatrix schreibt man nun, stellvertretend für die Kosten, die Entfernung in Kilometern. Dabei gibt das Element A1 die Entfernung von Lager 1 zu Filiale A an, also die Entfernung zwischen Bremen und Frankfurt (466). Das Element G3 gibt zum Beispiel die Entfernung vom Lager in Hannover zur Filiale in Mannheim an. Sind Lagerort und Filialenort identisch, so entstehen keine Transportkosten, daß entsprechend Element ist Null, so zum Beispiel I4 oder C2. Die fertig eingegebene Kostenmatrix sieht dann so aus: siehe Bild unten

	1	2	3	4	5
A	466	119	125	580	671
B	249	385	250	465	436
C	495	0	154	609	700
D	143	626	483	241	82
E	588	93	247	702	793
F	189	422	287	405	376
G	88	571	428	277	142
H	395	782	639	167	220
I	222	609	466	0	187
J	202	685	542	395	226
K	116	512	369	108	164
L	272	262	110	428	559

Fertig eingegebene Kostenmatrix.

Man kann sie nun abspeichern und hat sie dann jederzeit wieder zur Verfügung, und man kann davon ausgehen, daß sie sich im Laufe der Zeit nur wenig ändert. Ist bei der Eingabe ein Fehler unterlaufen, so wird durch Drücken von »3« ins Korrektur-Unterprogramm gesprungen.

Sieht die Matrix aus wie oben, wählt man Taste »5« und gibt erst die fünf Vorratsmengen, dann die 12 benötigten Mengen ein. Bei beiden Summen muß sich der Wert 589 ergeben. Ist keine Korrektur nötig, drückt man »N« und wählt dann »6«, um den Rechenvorgang zu starten. In der Tabelle, deren Begrenzung durch die grünschwarzen Balken markiert wird, erscheint zunächst die vorher eingegebene Kostenmatrix. In der sechsten Spalte stehen die Bedarfsmengen, in der letzten abgesetzten Zeile die Vorratsmengen. Das »Kreuzelement« rechts unten ist die Gesamtsumme der zu transportierenden Einzelmengen. Man kann die Zeile so lesen: Die Entfernungen beziehungsweise die Transportkosten nach Köln von Frankfurt betragen 189, von Hamburg 422, von Hannover 287, von Nürnberg 405 und von Stuttgart 376. Die benötigte Menge der Ware in Köln ist 71 Einheiten. Die Spalte 2 läßt sich so lesen: Die Kosten von Hamburg nach Bremen ... Bielefeld betragen 119 ... 262. Die vorrätige Menge in Hamburg sind 133 Einheiten.

Nach dem Gong erscheint die Lösung auf dem Bildschirm, die folgendermaßen aussehen müßte (Bild 3):

Gelesen wird das Lösungsbild so:
Lager 1 liefert an Filiale B 40, an D 6, an F 71, an G 35, an I 4, An J 13 und an K 8 Einheiten.
Lager 2 liefert an Filiale A 40, an B 16, an C 60, an E 17 Einheiten.

Lager 5 liefert nur an Filiale H und zwar 59 Einheiten. Eine Überprüfung der Zeilensummen beziehungsweise der Spaltensummen bestätigt die Richtigkeit des Ergebnisses. Die minimalen Gesamtkosten belaufen sich auf 80748. Die Rechenzeit bei diesem Beispiel beträgt zirka zirka 90 Sekunden.

(Daniel Rüster)

Lösung:

	1	2	3	4	5	
A	0	40	0	0	0	40
B	40	16	34	0	0	90
C	0	60	0	0	0	60
D	6	0	0	0	0	6
E	0	17	0	0	0	17
F	71	0	0	0	0	71
G	35	0	0	0	0	35
H	0	0	0	0	59	83
I	4	0	0	24	0	66
J	13	0	0	62	0	13
K	8	0	0	0	0	8
L	0	0	0	0	0	100
	177	133	134	86	59	589

Die fertig berechnete Kostenmatrix.

Bankcomputer

Zinsberechnung ist keine Zauberei: Wenn sie ihre Gewinne (oder Verluste) aus Prämiensparverträgen oder anderen Vermögensanlagen berechnen wollen, brauchen Sie nur das folgende Programm in Ihren TI-99/4A einzutippen.

TI 99/4A

```
        EINGABE

    S-PRAEMIENSPAREN

MONATLICHE RATE..:      50.00

ANSPARZEIT..(JHR):          6

FESTLEGUNG..(JHR):          7

S-PRAEMIENSATZ.%.:      14.00

ZINSSATZ.......%.:       5.00
```

Hardcopy einer Eingabemaske

Nachdem das Programm »Bankcomputer« gestartet wurde, erscheint das Hauptmenü. Geben Sie nun die gewünschte Funktion ein und betätigen Sie die »ENTER« Taste. Anschließend erscheint die Eingabemaske. Einige Standardwerte sind bereits eingetragen. Möchten Sie die Werte übernehmen, brauchen Sie an den entsprechenden Stellen nur die »ENTER« Taste zu drücken. Natürlich können diese Werte auch überschrieben, oder im Programm selbst verändert werden. Das erspart spätere Tipparbeit.

Alle Eingaben werden überprüft, ob sie innerhalb bestimmter Unter- und Obergrenzen liegen. Sollte Ihre Eingabe nicht stimmen, wird in Zeile 23 und 24 auf dem Bildschirm der Hinweis »Eingabe unzulässig«, »Min:x Max:y« ausgegeben. Diese vorgegebenen Werte können Sie jedoch in den entsprechenden Data-Zeilen im Programm ebenfalls verändern, wenn Ihnen zum Beispiel die Laufzeit eines Sparvertrages zu kurz sein sollte. Sind alle Werte eingegeben, erscheint am Bildschirm der Hinweis »Pw=1 / Gs=2 / Hc=3 / Rechnen= Enter«. Dabei steht Pw für Programmwahl, Gs für Grundstellung und Hc für Hardcopy. Nach Betätigung der Enter Taste wird die Berechnung ausgeführt und das Ergebnis angezeigt. In Zeile 24 erscheint dann wieder der Hinweis »Pw=1 / Gs=2 / Hc=3«. Durch Eingabe einer 1 gelangen Sie wieder zum Hauptmenü, mit der 2 direkt in die Eingabeseite und mit 3 erhalten Sie eine Hardcopy des Bildschirms.

Innerhalb der Eingabemaske können Sie den Cursor durch Eingabe eines negativen Wertes wieder nach oben bewegen.

Zum Schluß noch einige Erklärungen zu Begriffen die im Programm »Bankcomputer« auftauchen:

Kapitalisierung: Der Zeitpunkt, zu dem die angefallenen Zinsen zum Kapital addiert werden. In der Regel geschieht dies einmal im Jahr.

Ansparzeit/Festlegung: Die Ansparzeit ist die Zeitspanne, in der Sie Ihre Beiträge zahlen. Danach erfogt eine Wartezeit, in der das Kapital ruht und auch keine Beiträge geleistet werden. Ansparzeit plus Wartezeit ergibt die Festlegungsfrist.

(Werner Schäfer)

```
        ERGEBNIS

    S-PRAEMIENSPAREN

MONATLICHE RATE..:      50.00

GEZ. IN  6 JAHREN:    3600.00

S-PRAEMIE.14.00%.:     504.00

ZINSEN.... 5.00%.:     801.26

ENDKAPITAL.....:      4905.26
(NACH  7 JAHREN)

GEWINN.........:      1305.26
```

Die berechneten Zinsen werden auch in einer Maske ausgegeben. Hier das Beispiel einer Hardcopy

```
10 !=======================
20 !DER BANKCOMPUTER
30 !
40 !C. BY WERNER SCHAEFER
50 !
60 !TI-99/4A EXT. BASIC
70 !=======================
80 CALL CLEAR :: CALL SCREEN
(11)
90 DIM DEZ(12),TAFNR(12),TAE
KZ(12),TAPOS(12),TAZEI(12),T
ALANG(12),TAMASK(12)
100 DIM TAUN(12),TAOB(12)
110 IMAGE ########.##
120 IMAGE ###########
130 IMAGE "##.##%.:"
140 IMAGE ##.##
150 IMAGE ###
160 !====================
170 ! PROGRAMM WAHL
180 !====================
190 CALL CLEAR
200 RESTORE 2020 !ADRESSE TE
XTE
210 GOSUB 1640 !PROGRAMM WAH
L ANZEIGEN
220 ON WARNING NEXT
230 ACCEPT AT(15,12)SIZE(-1)
VALIDATE("1234")BEEP:PW
240 ON PW GOTO 270,700,1050,
250
250 STOP
260 !=============
270 ! FUNKTION 1
280 !=============
290 CALL CLEAR
300 RESTORE 2160 !ADRESSE EI
NGABE TEXTE
310 GOSUB 1640 !TEXTE ANZEIG
EN
320 FOR I=1 TO 12 :: DEZ(I)=
0 :: NEXT I :: I=1
330 DEZ(2)=5 :: DEZ(4)=1 !ST
ANDARDWERTE ZINSSATZ/KAPITAL
ISIERUNG
340 RESTORE 2300 :: GOSUB 17
40
350 I=1 :: GOSUB 1460 !UP EI
NGABE
360 GOSUB 1810 !UP TASTEN AB
FRAGE
370 ON KEY GOTO 410,160,290
380 !
390 !BERECHNUNG
400 !
410 IF DEZ(4)=3 THEN DEZ(4)=
4 ELSE IF DEZ(4)=4 THEN DEZ(
4)=12
420 F1=(DEZ(2)/(100*DEZ(4))+
1)
430 F2=INT((DEZ(3)/12))*DEZ(
4)
440 DEZ(5)=DEZ(1)*(F1^F2)
450 RM=DEZ(3)-(INT(DEZ(3)/12
)*12)
460 IF DEZ(4)=1 THEN DEZ(7)=
12 ELSE IF DEZ(4)=2 THEN DEZ
(7)=6 ELSE IF DEZ(4)=4 T
HEN DEZ(7)=3 ELSE IF DEZ(4)=
12 THEN DEZ(7)=1
470 IF DEZ(7)=12 THEN 520
480 IF RM=0 THEN 570
490 IF RM<=DEZ(7)THEN 520
500 DEZ(8)=DEZ(7)
510 GOSUB 530 :: RM=RM-DEZ(7
):: GOTO 480
520 DEZ(8)=RM :: GOSUB 530 :
: GOTO 570
530 !
540 DEZ(9)=(((DEZ(8)*DEZ(2))
/1200)+1)
```

```
550 DEZ(5)=DEZ(5)*DEZ(9)
560 RETURN
570 !
580 DEZ(6)=DEZ(5)-DEZ(1)
590 !
600 !AUSGABE ERGEBNIS
610 !
620 I=1 :: CALL CLEAR
630 RESTORE 2370 !ADRESSE AU
SGABE-TEXTE
640 GOSUB 1640 !UP TEXT AUSG
EBEN
650 RESTORE 2450 !ADRESSE DE
Z.FELD BESCHREIBUNG
660 GOSUB 1700 !UP DEZ. FELD
 AUSGEBEN
670 GOSUB 1780 !TASTEN ABFRA
GE
680 ON KEY GOTO 670,160,290
690 !==============
700 ! FUNKTION 2
710 !==============
720 CALL CLEAR
730 RESTORE 2570 !ADRESSE EI
NGABE TEXTE
740 GOSUB 1640 !TEXTE ANZEIG
EN
750 FOR I=1 TO 12 :: DEZ(I)=
0 :: NEXT I
760 DEZ(5)=5.00 :: DEZ(2)=4
!STANDARDWERTE ZINSSATZ/EINZ
AHLUNGS INTERVALL
770 RESTORE 2670 :: I=1 :: G
OSUB 1740 !DEZ FELDER ANZEIG
EN
780 I=1 :: GOSUB 1460 !UP EI
NGABE
790 GOSUB 1830 !TASTENABFRAG
E
800 ON KEY GOTO 840,160,720
810 !
820 !BERECHNUNG
830 !
840 IF DEZ(2)=3 THEN DEZ(2)=
4 ELSE IF DEZ(2)=4 THEN DEZ(
2)=12
850 F1=(DEZ(5)*((DEZ(2)+1)/2
00))+DEZ(2)
860 F2=((DEZ(5)/100)+1)
870 F3=(DEZ(3)/12)
880 F4=F2^F3-1
890 F5=(F1*100*DEZ(1)*F4)/DE
Z(5)
900 F4=F2^F3
910 F1=DEZ(4)*F4
920 DEZ(10)=F1+F5
930 DEZ(9)=DEZ(1)*F3*DEZ(2)
940 DEZ(8)=DEZ(10)-DEZ(9)
950 !
960 ! ERGEBNIS AUSGABE
970 !
980 CALL CLEAR
990 RESTORE 2760 !ADRESSE ER
GNIS TEXTE
1000 GOSUB 1640 !UP TEXT AUS
GABE
1010 RESTORE 2850 !ADRESSE D
EZ.FELD BESCHREIBUNG
1020 I=1 :: GOSUB 1700 !DEZ.
FELD AUSGABE
1030 GOSUB 1780 !TASTENABFRA
GE
1040 ON KEY GOTO 1030,160,72
0
1050 !
1060 !===============
1070 ! FUNKTION 3
1080 !==============
1090 !
1100 CALL CLEAR
1110 RESTORE 3000 !ADRESSE E
/TEXTE
1120 GOSUB 1640 !TEXTE ANZEI
GEN
1130 FOR I=1 TO 12 :: DEZ(I)
=0 :: NEXT I :: I=1
1140 DEZ(2)=6 :: DEZ(3)=7 ::
 DEZ(4)=14.00 :: DEZ(5)=5 !S
TANDARDWERTE ANSPARZEIT/
FESTLEGUNG/PRAEMIENSATZ/ZINS
SATZ
1150 RESTORE 3120 ! ADRESSE
DEZFELD BESCHREIBUNG
1160 GOSUB 1740
1170 I=1 :: GOSUB 1460 !EING
ABE
1180 GOSUB 1810 !TASTEN ABFR
AGE
1190 ON KEY GOTO 1200,160,11
00
1200 !
1210 ! BERECHNUNG
1220 !
1230 DEZ(6)=DEZ(1)*12*DEZ(2)
1240 DEZ(7)=((DEZ(6)*DEZ(4))
/100)
1250 Q=((DEZ(5)/100)+1)
1260 F1=((12/(Q-1))+6.5)
1270 F2=Q^DEZ(2)-1
1280 F3=Q^(DEZ(3)-DEZ(2))
1290 DEZ(8)=((DEZ(1)*F1*F2*F
3)-DEZ(6))
1300 DEZ(9)=DEZ(6)+DEZ(7)+DE
Z(8)
1310 DEZ(10)=DEZ(9)-DEZ(6)
1320 !ERGEBNIS
1330 !
1340 I=1 :: CALL CLEAR
1350 RESTORE 3220 ! ADRESSE
A TEXTE
1360 GOSUB 1640 ! ANZEIGE
1370 RESTORE 3360 ! ADR. DEZ
F.
1380 GOSUB 1700 ! DEZ.FELDAU
SGABE
1390 GOSUB 1780 ! TASTEN.ABF
RAGE
1400 ON KEY GOTO 1390,160,11
00
1410 !
1420 ! UNTERPROGRAMME
1430 !
1440 !UP EINGABE
1450 !
1460 ON WARNING NEXT
1470 ACCEPT AT(TAZEI(I),TAPO
S(I))SIZE(TALANG(I))BEEP VAL
IDATE(NUMERIC,"-"):EIN
1480 EIN$=STR$(EIN):: PO=POS
(EIN$,"-",1):: IF PO=0 THEN
1520
1490 GOSUB 1930
1500 IF I<>1 THEN I=I-1
1510 GOTO 1460
1520 !
1530 IF EIN<TAUN(I)OR EIN>TA
OB(I)THEN 1590
1540 DEZ(TAFNR(I))=EIN
1550 GOSUB 1930
1560 IF TAEKZ(I)>1 THEN RETU
RN
1570 I=I+1
1580 GOTO 1460
1590 CALL SOUND(800,400,0)::
 DISPLAY AT(23,1):"EINGABE U
NZULAESSIG"
1600 DISPLAY AT(24,1):"MIN:"
;TAUN(I);"MAX:";TAOB(I)
1610 CALL KEY(0,KEY,ST):: IF
 ST=0 THEN 1610
1620 DISPLAY AT(23,1):" " ::
 DISPLAY AT(24,1):" " :: GOT
O 1460
1630 !
1640 ! UP TEXTE ANZEIGEN
1650 !
1660 READ ZEI,POSI,TEXT$
1670 IF ZEI=99 THEN RETURN
1680 DISPLAY AT(ZEI,POSI)SIZ
E(LEN(TEXT$)):TEXT$
1690 GOTO 1640
1700 !
1710 ! DEZIMALFELDER ANZEIGE
N
1720 !
1730 I=I+1
1740 READ TAEKZ(I),TAFNR(I),
TAZEI(I),TAPOS(I),TALANG(I),
TAMASK(I),TAUN(I),TAOB(I
)
1750 GOSUB 1930
1760 IF TAEKZ(I)=2 THEN 1770
 ELSE 1730
1770 RETURN
1780 !
1790 DISPLAY AT(24,1):"PW=1/
GS=2/HC=3" :: GOTO 1840
1800 !
1810 ! UP TASTEN ABFRAGE
1820 !
1830 DISPLAY AT(24,1):"PW=1/
GS=2/HC=3/ENTER=RECHNEN"
1840 CALL KEY(0,KEY,ST):: IF
 ST=0 THEN 1840
1850 IF KEY=13 THEN KEY=1 ::
 RETURN
1860 IF KEY=49 THEN KEY=2 ::
 RETURN
1870 IF KEY=50 THEN KEY=3 ::
 RETURN
1880 IF KEY=51 THEN GOSUB 29
90
1890 GOTO 1840
1900 !
1910 ! UP FELDANZEIGE
1920 !
1930 IF TAMASK(I)=5 THEN 199
0 ELSE IF TAMASK(I)=4 THEN 1
980 ELSE IF TAMASK(I)=3
THEN 1970 ELSE IF TAMASK(I)=
2 THEN 1960
1940 IF TAMASK(I)=0 THEN 200
0
1950 DISPLAY AT(TAZEI(I),TAP
OS(I)):USING 110:DEZ(TAFNR(I
)):: RETURN
1960 DISPLAY AT(TAZEI(I),TAP
OS(I)):USING 120:DEZ(TAFNR(I
)):: RETURN
1970 DISPLAY AT(TAZEI(I),TAP
OS(I)):USING 130:DEZ(TAFNR(I
)):: RETURN
1980 DISPLAY AT(TAZEI(I),TAP
OS(I)):USING 140:DEZ(TAFNR(I
)):: RETURN
1990 DISPLAY AT(TAZEI(I),TAP
OS(I)):USING 150:DEZ(TAFNR(I
)):: RETURN
2000 DISPLAY AT(TAZEI(I),TAP
OS(I))SIZE(TALANG(I)):DEZ(TA
```

Listing zum Programm » Bankcomputer «

```
FNR(I)):: RETURN
2010 !
2020 ! PROGRAMM UEBERSICHT
2030 !
2040 DATA 1,5,DER BANKCOMPUT
ER
2050 DATA 2,5,==============
==
2060 DATA 5,0,ENDKAPITAL
2070 DATA 6,5,BEI EINMALZAHL
UNG...1
2080 DATA 7,5,BEI PERIODENZA
HLUNG.2
2090 DATA 9,0,S-PRAEMIENSPAR
EN.....3
2100 DATA 11,0,ENDE.........
..........4
2110 DATA 15,0,IHRE WAHL.?
2120 DATA 99,99,ENDE
2130 !
2140 ! EINGABE FUNKTION 1
2150 !
2160 DATA 1,9,EINGABE
2170 DATA 3,0,ENDKAPITAL BEI
 EINMALZAHLUNG
2180 DATA 7,0,EINZAHLUNG...D
M..:
2190 DATA 9,0,ZINSSATZ.....%
...:
2200 DATA 11,0,ANLAGEDAUER.(
MON):
2210 DATA 13,0,KAPITALISIERU
NG..:
2220 DATA 14,0,(1=JHL/2=HJHL
/3=VJHL/4=MON.)
2230 DATA 99,99,ENDE
2240 !
2250 ! FELD-BESCHREIBUNG
2260 !
2270 ! ENDE-KZ,DEZIMALFELD-N
R,ZEILE,POSITION,LAENGE,MASK
EN-NR,UNTER,-OBERGRENZE
2280 !
2290 ! EINZAHLUNG
2300 DATA 0,1,7,19,12,1,1.00
,9999999.99
2310 ! ZINSSATZ
2320 DATA 0,2,9,24,-5,4,1.00
,99.99
2330 ! ANLAGEDAUER
2340 DATA 0,3,11,26,3,5,1,24
0
2350 ! KAPITALISIERUNG
2360 DATA 2,4,13,27,-2,0,1,4
2370 !ERGEBIS TEXTE FUNKTION
 1
2380 DATA 1,9,ERGEBNIS
2390 DATA 3,0,ENDKAPITAL BEI
 EINMALZAHLUNG
2400 DATA 7,0,EINZAHLUNG...D
M.:
2410 DATA 9,0,+ZINSEN.......
..:
2420 DATA 11,0,ENDKAPITAL...
DM.:
2430 DATA 13,0,(NACH      MON
ATEN)
2440 DATA 99,99,END
2450 !ERGEBNISFELDER FUNKTIO
N 1
2460 !EINZAHLUNG
2470 DATA 0,1,7,19,9,1,0,0
2480 !ZINSSATZ
2490 DATA 0,2,9,10,0,3,0,0
2500 !ZINSEN DM
2510 DATA 0,6,9,19,12,1,0,0
2520 !
2530 DATA 0,5,11,19,12,1,0,0
2540 DATA 2,3,13,06,-4,0,0,0
2550 !
2560 ! EINGABE FUNKTION 2
2570 DATA 1,9,EINGABE
2580 DATA 3,0,ENDKAPITAL PER
IODENZAHLUNG
2590 DATA 7,0,PERIOD.EINZAHL
UNG:
2600 DATA 9,0,EINZAHLG. PERI
ODE:
2610 DATA 10,0,(1=JHL/2=HJHL
/3=VJHL/4=MON)
2620 DATA 12,0,ANLAGEDAUER.(
MON):
2630 DATA 14,0,ANFANGSKAPITA
L...:
2640 DATA 16,0,ZINSSATZ.....
..%.:
2650 DATA 99,99,ENDE
2660 !
2670 DATA 0,1,7,19,10,1,1.00
,99999.99
2680 ! EINZAHLUNGS/PERIODE
2690 DATA 0,2,9,27,-2,0,1,4
2700 ! ANLAGEDAUER
2710 DATA 0,3,12,26,03,5,1,2
40
2720 ! ANFANGSKAPITAL
2730 DATA 0,4,14,19,10,1,0.0
0,9999999.99
2740 ! ZINSSATZ
2750 DATA 2,5,16,24,-5,4,1.0
0,99.99
2760 DATA 1,9,ERGEGNIS
2770 DATA 3,0,ENDKAPITAL PER
IODENZAHLUNG
2780 DATA 7,0,PERIOD.EINZAHL
UNG:
2790 DATA 9,0,GEZ.IN      MON
ATEN
2800 DATA 11,0,ZINSEN.......
....:
2810 DATA 13,0,ANFANGSKAPITA
L...:
2820 DATA 15,0,ENDKAPITAL...
....:
2830 DATA 99,99,ENDE
2840 !P EINZ.
2850 DATA 0,1,7,19,12,1,0,0
2860 ! MON
2870 DATA 0,3,9,7,-4,0,0,0
2880 ! GEZ IN
2890 DATA 0,9,9,19,12,1,0,0
2900 !ZINSSATZ
2910 DATA 0,5,11,11,0,3,0,0
2920 ! ZINSEN
2930 DATA 0,8,11,19,12,1,0,0
2940 ! A KAP
2950 DATA 0,4,13,19,12,1,0,0
2960 ! E KAP
2970 DATA 2,10,15,19,12,1,0,
0
2980 !
2990 !
3000 DATA 1,12,EINGABE
3010 DATA 3,7,S-PRAEMIENSPAR
EN
3020 DATA 7,0,MONATLICHE RAT
E..:
3030 DATA 9,0,ANSPARZEIT..(J
HR):
3040 DATA 11,0,FESTLEGUNG..(
JHR):
3050 DATA 13,0,S-PRAEMIENSAT
Z.%.:
3060 DATA 15,0,ZINSSATZ.....
..%.:
3070 DATA 99,99,ENDE
3080 !
3090 ! EINGABE FUNKTION 3
3100 !
3110 !RATE
3120 DATA 0,1,7,19,12,1,1.00
,9999.99
3130 !ANSPARZEIT
3140 DATA 0,2,9,27,-2,0,4,7
3150 !FESTLEGUNG
3160 DATA 0,3,11,27,-2,0,7,9
3170 !S-PRAEMIENSATZ
3180 DATA 0,4,13,24,-05,4,5.
00,25.00
3190 !ZINSSATZ
3200 DATA 2,5,15,24,-05,4,1.
00,99.99
3210 !
3220 DATA 1,12,ERGEBNIS
3230 DATA 3,7,S-PRAEMIENSPAR
EN
3240 DATA 7,0,MONATLICHE RAT
E..:
3250 DATA 9,0,GEZ. IN     JAH
REN:
3260 DATA 11,0,S-PRAEMIE....
. % :
3270 DATA 13,0,ZINSEN.......
..%.:
3280 DATA 15,0,ENDKAPITAL...
....:
3290 DATA 16,0,(NACH      JAHR
EN)
3300 DATA 18,0,GEWINN.......
....:
3310 DATA 99,99,ENDE
3320 !
3330 ! ERGEBNIS FU 3
3340 !
3350 ! RATE
3360 DATA 0,1,7,19,8,1,0,0
3370 ! ANSP. JAHRE
3380 DATA 0,2,9,09,-2,0,0,0
3390 ! ANGESP. GUTHABEN
3400 DATA 0,6,9,19,12,1,0,0
3410 !S-P SATZ
3420 DATA 0,4,11,11,0,3,0,0
3430 ! S-P DM
3440 DATA 0,7,11,19,12,1,0,0
3450 ! ZINS.SATZ
3460 DATA 0,5,13,11,0,3,0,0
3470 ! ZINSEN
3480 DATA 0,8,13,19,8,1,0,0
3490 ! ENDKAPITAL
3500 DATA 0,9,15,19,8,1,0,0
3510 ! E JAHRE
3520 DATA 0,3,16,7,-2,0,0,0
3530 !GEWINN
3540 DATA 2,10,18,19,12,1,0,
0
3550 !
3560 ! HARDCOPY ROUTINE
3570 ! (HIER EPSON RX80/V.24
)
3580 !
3590 OPEN #1:"RS232.BA=600.D
A=8.PA=E.CH"
3600 FOR Z=1 TO 20
3610 FOR P=1 TO 30
3620 CALL GCHAR(Z,P,ZEI)
3630 DRU$=SEG$(DRU$,1,P)&CHR
$(ZEI)
3640 NEXT P
3650 PRINT #1:DRU$
3660 NEXT Z
3670 CLOSE #1
3680 RETURN
```

Listing zum Programm » Bankcomputer « (Schluß)

TIPS&TRICKS

Sort ohne Längenprüfung für Sharp

Bekanntlich macht das Basic SA-5510 des MZ-80A von Sharp Schwierigkeiten beim Sortieren von Zeichenketten. Es werden nur dann korrekt alphabetische Vergleiche angestellt, wenn die zu sortierenden Strings von gleicher Länge sind. Aus dieser Notlage kann man sich durch geeignete Programmierung befreien; in der Regel verlängert das jedoch den Sortiervorgang. Oder man füllt die zu ordnenden Strings mit Leerzeichen auf, bringt sie so auf einheitliche Länge und vergeudet damit Speicherplatz.

Wenn man dagegen darauf verzichtet, daß beim Sortieren auch die Länge jedes Strings Beachtung findet, dann genügt ein kurzer Maschinenbefehl, um zu einer zufriedenstellenden Lösung zu kommen. Mit der Anweisung $2305 C3 0C 23 oder von Basic aus mit POKE 8965,195:POKE 8966,12:POKE 8967,35 wird der Längenvergleich im Interpreter übersprungen.

Mit $2305 78 B1 C8 oder mit POKE 8965,120:POKE 8966,177:POKE 8967,200 kann der ursprüngliche Zustand wieder erreicht werden. (W. Müller)

TRS-80 Modell I/III — Schnelle Grafik in Basic

Wenn Sie jemals versucht haben, in Basic ein schnelles Grafikprogramm zu schreiben, werden Sie festgestellt haben, daß das unmöglich ist. Wenn Sie eine etwas größere Erfahrung mit Computern haben, kennen Sie bestimmt die String-Pack-Technik, die erstmals von Leo B. Christopherson angewendet wurde. Mit dieser Technik sind viele TRS-80-Besitzer in die Welt der schnellen Grafik vorgestoßen. Diese Methode erzeugt eine der schnellsten Grafiken, die von Basic aus möglich ist.

Es gibt jedoch noch eine Alternative, zum Beispiel für einen Programmvorspann oder für Displays, um Aufmerksamkeit zu erregen. Ich wurde auf diese Möglichkeit aufmerksam, als ich mit dem 32-Zeichen-Mode arbeitete.

Um von der schnellen Grafik Gebrauch zu machen, muß das Programm die Bildschirmanzeige in den 32-Zeichen-Mode bringen und die gewünschten Zeichen in die ungeraden Speicherstellen POKEn oder PRINTen. Nun sind die Zeichen noch unsichtbar im Bildschirmspeicher, und wenn man in den 64-Zeichen-Mode umschaltet, werden sie mit (für Basic) unglaublicher Geschwindigkeit am Bildschirm angezeigt.

Das Beste an dieser Technik ist, daß man üblicherweise Speicherplatz spart und trotzdem durch ein blinkendes Menü, eine Grafik und so weiter die Aufmerksamkeit erweckt. Hier ist ein kurzes Beispielprogramm:

```
10 CLS:PRINTCHR$(23)
20 FOR X=1 TO 1023 STEP 2:POKE 15360+X,191:NEXT X
30 FOR X=0 TO 1023 STEP 6:POKE 15360+X,191:NEXT X
40 FOR T=1 TO 9:FOR Z=1 TO 50
50 OUT 255,8:FOR R=1 TO Z:NEXT R
60 OUT 255,0:FOR R=1 TO Z:NEXT R
70 NEXT Z,T
```

Wenn Sie mit einem TRS-80 MIII arbeiten, müssen Sie alle 255 in 236 und die Zahlen 8 und 0 in 4 und 0 ändern.

(G. Hornung)

TI 99/4A — Arrays schnell auf Kassette speichern

Zwei Methoden sind praktikabel:

1) Man schreibt die entsprechenden Daten ins Minimem entweder mit LOAD oder indem man das Minimem als File eröffnet. Die Daten können nach Beendigung des Programms mit Easy-Bug durch den Befehl S7000 TO 7FFF auf Kassette gespeichert werden. Das Einlesen der Daten erfolgt wieder mit Easy-Bug und Kommando L.

Nachteil dieser Methode: Das Basic-Programm geht verloren, da man QUIT drücken muß, bevor man zu Easy-Bug kommt.

2) Man schreibt die Daten durch POKEV ins VDP-RAM und speichert sie gemeinsam mit dem Basic-Programm auf Kassette. Dazu eine Erläuterung: Normalerweise hat RAMTOP den Wert 16383 = 256.63+255. Bei Verwendung eines Diskettensystems wird dieser Wert entsprechend vermindert. Man kann nun ein Diskettensystem vortäuschen, indem man auf Adresse -31888 einen entsprechend niedrigeren Wert hineinlädt. Dadurch gewinnt man einen Speicherplatz, in den man Werte ablegen kann, um sie später auf Kassette zu speichern. Dazu muß man folgendermaßen vorgehen:

```
⟩CALL LOAD (-31888,62,255)
⟩NEW
```

Jetzt kann ein Basic-Programm eingegeben werden, zum Beispiel:

```
100   CALL PEEK (-31888,K,W)
110   PRINT "RAMTOP IST";256*K+W
120   INPUT "DATEN SCHREIBEN/LESEN = ⟩L/S":I$
130   FOR K=16300 TO 16383
140   IF I$◊"S"THEN 180
150   W=K-16300
160   CALL POKEV(K,W)
170   GOTO 190
180   CALL PEEKV(K,W)
190   PRINT K,W
200   NEXT K
```

Dieses Programm meldet zuerst, daß RAMTOP gleich 16127 ist. Durch Eingabe von S werden im VDP-RAM oberhalb des Basic-Programms die Zahlen von 0 bis 83 gespeichert. Will man diese Werte auf Kassette speichern, muß zuerst RAMTOP wieder auf 16383 gesetzt werden, da sonst nur das Basic-Programm, nicht aber die Daten abgespeichert werden!

```
⟩CALL LOAD(-31888,63,255)
⟩SAVE CS1
```

Will man die Daten wieder lesen, dann gibt man

```
⟩OLD CS1
⟩CALL LOAD(-31888),62,255
⟩RUN
```

ein und wählt diesmal L für Lesen. (Natürlich jetzt nicht mehr NEW eingeben, sonst ist das Basic-Programm weg!) Wenn man RAMTOP entsprechend herabsetzt, können mit dieser Methode große Zahlenmengen gespeichert werden, man muß jedoch genau Buch führen, wohin man die einzelnen Daten gelegt hat.

Auch diese Methode benötigt wegen PEEKV und POKEV zum Beispiel das Mini-Memory-Modul.

(H. Reidlinger)

Eingabe-masken automatisch erstellen

Der TI 99/4A schreibt sich selbst ein Daten-Eingabe-Programm.

Nach dem Start des Programms kann mit Hilfe der Cursor-Tasten eine Maske auf dem Bildschirm erstellt werden. Es ist zu beachten, daß die Stellen, an denen Variablen eingegeben werden müssen, mit einem Doppelkreuz (#) gekennzeichnet werden. Pro Bildschirmzeile kann aber immer nur eine Variable eingegeben werden. Die Länge der Variablen wird durch die Anzahl der Doppelkreuze bestimmt.

Haben Sie die Maske in ihrer endgültigen Form auf dem Bildschirm entworfen, muß die »ENTER«-Taste betätigt werden. Die Datensätze werden anschließend berechnet. Während dieser Zeit wird auf der linken Bildschirm-Hälfte angezeigt, daß der Computer arbeitet, es können in dieser Zeit also keine Eingaben gemacht werden.

Anschließend wird man nach dem Dateinamen des zu generierenden Programms gefragt. Dann müssen noch die Namen der einzelnen Felder eingegeben werden, wie zum Beispiel: VORNAME$, NAME$ und ORT$. Die Bezeichnungen werden dann, nach Eingabe der Zeilennummer, ab der das erzeugte Programm numeriert werden soll, übernommen und auf Diskette abgespeichert.

Nach Betätigung der»ENTER«-Taste kann die erzeugte Maske mit dem »MERGE«-Befehl an bereits bestehende Programme angefügt werden. (Volker Küsters)

Listing zum »Maskengenerator«

```
100 !*************************************
*************************************
110 !*                                   *
120 !*          PROGRAMMGENERATOR FUER D
ATENEINGABE UEBER TASTATUR               *
130 !*                                   *
140 !*  HARDWAREKONFIGURATION: TI 99/4A
    X-BASIC      DISKETTENLAUFWERK       *
150 !*                                   *
160 !* JANUAR 84            VOLKER KUESTE
RS/56 WUPPERTAL-11/ELFENHANG 5           *
170 !*                                   *
180 !*************************************
*************************************
190 REM HAUPTPROGRAMM
200 DIM SATZ$(48),VH(24),VV(24),VAR$(24)
,SV(24),DISP$(24)
210 CALL MASKE
220 CALL DISPLAY(SATZ$(),VH(),VV(),DISP$
())
230 CALL EINGABE(SATZ$(),VV(),VAR$(),SV(
),DISP$(),FILE$)
240 CALL ACCEPT(SATZ$(),VV(),VH(),VAR$()
,SV())
250 CALL SPEICHERN(SATZ$(),FILE$)
260 END
270 ! HAUPTPROGRAMM ENDE ***************
*************************************
280 SUB MASKE
290 CALL CHAR(129,"FF0101010101010101")! C
URSOR DEFINIEREN
300 CALL CLEAR
310 CALL SPRITE(#1,129,2,1,17)
320 CALL POSITION(#1,Z,S)! CURSOR SETZEN
330 CALL KEY(0,KEY,STATUS)
340 IF STATUS=0 THEN 330
350 IF KEY=11 THEN 360 ELSE 380 ! PFEIL
OBEN
360 IF Z>1 THEN Z=Z-8 ELSE 320
370 GOTO 520
380 IF KEY=10 THEN 390 ELSE 410 ! PFEIL
UNTEN
390 IF Z<185 THEN Z=Z+8 ELSE 320
400 GOTO 520
410 IF KEY=9 THEN 420 ELSE 440 ! PFEIL R
ECHTS
420 IF S<233 THEN S=S+8 ELSE 320
430 GOTO 520
440 IF KEY=8 THEN 450 ELSE 470 ! PFEIL L
INKS
450 IF S>17 THEN S=S-8 ELSE 320
460 GOTO 520
470 IF KEY=13 THEN 540 ! ENTER
480 CALL HCHAR(INT(Z/8)+1,INT(S/8)+1,KEY
)! CHARAKTER SETZEN
490 IF S=233 AND Z<185 THEN S=17 :: Z=Z+
8 :: GOTO 520 ! ZEILENENDE ?
500 IF S=233 AND Z=185 THEN 520 ! BILDSC
HIRM VOLL ?
510 S=S+8
520 CALL LOCATE(#1,Z,S)! CURSOR AUF NEUE
 POSITION SETZEN
530 GOTO 320
540 CALL DELSPRITE(#1)! CURSOR LOESCHEN
550 SUBEND
560 ! MASKE ENDE ***********************
*************************************
570 SUB DISPLAY(SATZ$(),VH(),VV(),DISP$(
))
580 FOR I=1 TO 24
590 CALL HCHAR(I,2,62)
600 FOR J=3 TO 30
610 CALL GCHAR(I,J,CHAR)
620 DISP$(I)=DISP$(I)&CHR$(CHAR)
630 NEXT J
640 NEXT I
650 FOR I=1 TO 24
660 CALL HCHAR(I,2,32)
670 VV(I)=POS(DISP$(I),"#",1)
680 VH(I)=VV(I)
690 IF VV(I)=0 THEN 750
700 FOR K=VV(I)+1 TO 28
710 IF SEG$(DISP$(I),K,1)<>"#" THEN 730
720 NEXT K
730 VH(I)=K
```

```
740 DISP$(I)=SEG$(DISP$(I),1,VV(I)-1)&RP
T$(" ",VH(I)-VV(I))&SEG$(DISP$(I),VH(I),
29-VH(I))
750 IF DISP$(I)=RPT$(" ",28)THEN 790
760 SATZ$(I)=CHR$(162)&CHR$(240)&CHR$(18
3)&CHR$(200)
770 IF I>9 THEN W=2 ELSE W=1
780 SATZ$(I)=SATZ$(I)&CHR$(W)&STR$(I)&CH
R$(179)&CHR$(200)&CHR$(1)&CHR$(49)&CHR$(
182)&CHR$(181)&CHR$(199)&CHR$(28)&DISP$(
I)&CHR$(0)
790 NEXT I
800 SUBEND
810 ! DISPLAY ENDE ********************
****************************************
820 SUB EINGABE(SATZ$(),VV(),VAR$(),SV()
,DISP$(),FILE$)
830 CALL CLEAR
840 DISPLAY AT(2,2):"DATEINAME ?"
850 ACCEPT AT(4,4)SIZE(15)BEEP:FILE$
860 DISPLAY AT(7,1):" GEBEN SIE DIE ZUR
NACHFOL-  GENDEN ZEILE GEHOERENDE     VA
RIABLE EIN"
870 FOR I=1 TO 24
880 IF VV(I)=0 THEN 920
890 DISPLAY AT(11,1):DISP$(I)
900 ACCEPT AT(13,4)SIZE(10)BEEP:VAR$(I)
910 IF SEG$(VAR$(I),LEN(VAR$(I)),1)="$"
THEN SV(I)=199 ELSE SV(I)=200
920 NEXT I
930 SUBEND
940 ! EINGABE ENDE ********************
****************************************
950 SUB ACCEPT(SATZ$(),VV(),VH(),VAR$(),
SV())
960 FOR I=25 TO 48
970 IF VV(I-24)=0 THEN 1050
980 SATZ$(I)=CHR$(164)&CHR$(240)&CHR$(18
3)&CHR$(200)
990 IF I-24>9 THEN W=2 ELSE W=1
1000 SATZ$(I)=SATZ$(I)&CHR$(W)&STR$(I-24
)&CHR$(179)&CHR$(200)
1010 IF VV(I-24)>9 THEN W=2 ELSE W=1
1020 SATZ$(I)=SATZ$(I)&CHR$(W)&STR$(VV(I
-24))&CHR$(182)&CHR$(235)&CHR$(183)&CHR$
(200)
1030 IF VH(I-24)-VV(I-24)>9 THEN W=2 ELS
E W=1
1040 SATZ$(I)=SATZ$(I)&CHR$(W)&STR$(VH(I
-24)-VV(I-24))&CHR$(182)&CHR$(238)&CHR$(
181)&VAR$(I-24)&CHR$(0)
1050 NEXT I
1060 SUBEND
1070 ! ACCEPT ENDE ********************
****************************************
1080 SUB SPEICHERN(SATZ$(),FILE$)
1090 CALL CLEAR
1100 DISPLAY AT(2,2):"ZEILENNUMMER ?"
```

```
1110 ACCEPT AT(4,4)SIZE(5)VALIDATE(NUMER
IC)BEEP:ZEILE
1120 DISPLAY AT(7,2):"INKREMENT ?"
1130 ACCEPT AT(9,4)SIZE(3)VALIDATE(NUMER
IC)BEEP:INK
1140 DISPLAY AT(14,1):"      ABSPEICHERN
FILE !"
1150 OPEN #1:FILE$,DISPLAY ,VARIABLE 163
,OUTPUT
1160 SATZ$(0)=CHR$(157)&CHR$(200)&CHR$(5
)&"CLEAR"&CHR$(0)
1170 FOR I=0 TO 48
1180 IF SATZ$(I)="" THEN 1220
1190 GOSUB 1260
1200 SATZ$(I)=CHR$(ZEI1)&CHR$(ZEI2)&SATZ
$(I)
1210 PRINT #1:SATZ$(I)
1220 NEXT I
1230 PRINT #1:CHR$(255)&CHR$(255)
1240 CLOSE #1
1250 GOTO 1300
1260 ZEI1=INT(ZEILE/256)
1270 ZEI2=ZEILE-(ZEI1*256)
1280 ZEILE=ZEILE+INK
1290 RETURN
1300 SUBEND
1310 ! SPEICHERN ENDE ****************
****************************************
```

```
BILDSCHIRMHARDCOPY

****************************
  DATENEINGABE   T-E-S-T
****************************
  NAME: ################
  DATUM:         ######
  ORT:      ############


  STRECKE A:   ######## [cm]
  STRECKE B:   ######## [cm]

  WINKEL (alpha): #####
```

Hardcopy eines Bildschirms beim Definieren der einzelnen Felder

◄

So sieht das erzeugte Merge-Programm aus (siehe hierzu auch die Hardcopy)

▼

```
100 CALL CLEAR
102 DISPLAY AT(1,1):"****************************"
104 DISPLAY AT(3,1):"  DATENEINGABE   T-E-S-T    "
106 DISPLAY AT(5,1):"****************************"
108 DISPLAY AT(7,1):"  NAME:                     "
110 DISPLAY AT(9,1):"  DATUM:                    "
112 DISPLAY AT(11,1):"  ORT:                     "
114 DISPLAY AT(15,1):"  STRECKE A:          [cm]"
116 DISPLAY AT(17,1):"  STRECKE B:          [cm]"
118 DISPLAY AT(20,1):"  WINKEL (alpha):          "
120 ACCEPT AT(7,9)SIZE(15)BEEP:NAME$
122 ACCEPT AT(9,18)SIZE(6)BEEP:DATE$
124 ACCEPT AT(11,12)SIZE(12)BEEP:PLACE$
126 ACCEPT AT(15,16)SIZE(8)BEEP:A
128 ACCEPT AT(17,16)SIZE(8)BEEP:B
130 ACCEPT AT(20,19)SIZE(5)BEEP:W
```

KAYPRO II
Der Portable Computer des Jahres 1983
mit Drucker
CP/M-Matrix-Drucker (Epson-komp.) DM 5995,-
Computer, Drucker, Druckerkabel und Software inkl. MwSt.
inkl. CP/M, Wordstar, Supercalc, MBasic, dBASE II, The Word Plus, Kaypro 4, Kaypro 10 auch ab Lager lieferbar
Fa. HMS
T. 089/295991 · Isartorplatz 4 · 8000 München 2

Kurven-diskussionen

Das Programm führt mit jeder beliebigen Funktion eine vollständige Kurvendiskussion aus, stellt sie grafisch dar, liefert eine Wertetabelle und berechnet die Fläche unter einem Graphen. Für diese Mathematikaufgabe wird ein TI 99/4A und die Extended-Basic-Erweiterung benötigt.

Der Computer erteilt selbst die Anweisungen, welche Daten einzugeben sind. Wenn das Menü erscheint, kann man eingeben, welche Operation durchgeführt werden soll, indem man die entsprechende Taste (1-8) drückt.

Mathematische Methoden:

Die Ableitungen der ganzrationalen Funktion 5. Grades sind bereits eingegeben (Zeilen 370 bis 400). Bei anderen Funktionen müssen die Ableitungen eingegeben werden. Die Nullstellen werden durch Nullstellenbestimmung der Stammfunktion, die Extremstellen und Wendestellen durch Nullstellenbestimmung der entsprechenden Ableitungen errechnet. Die Fläche unter dem Graphen wird nach der »Simpson«-Methode bestimmt.

Anmerkung zum Programmablauf:

Wenn man keine ganzrationale Funktion 5. Grades, sondern eine andere Funktion untersuchen lassen will, muß man die Ableitungen selbst eingeben. Dadurch gehen die Ableitungen der ganzrationalen Funktion verloren. Man sollte das Programm also vorher zum Beispiel auf Kassette speichern.

Wenn man eine grafisch dargestellte Funktion mit anderen Grenzen noch einmal darstellen lassen will, muß man nur die Gleichheitstaste drücken. Mit jeder anderen Taste springt man zum Menü zurück.

Die Wertetabelle kann man mit der Gleichheitstaste beenden und dann zum Menü zurückspringen.

(Michael Lyhs)

```
100 ! ********************
110 ! *KURVENDISKUSSION*
120 ! *       VON        *
130 ! *MICHAEL LYHS      *
140 ! *KARLSTR. 35       *
150 ! *6350 BAD NAUHEIM*
160 ! *TEL.:06032/6539 *
170 ! * FEBRUAR 1984    *
180 ! ********************
190 DIM J(30):: DIM JM(30):: DIM
    JN(30):: DIM JW(30)
200 KW=0 :: GOSUB 1580 :: DISPLA
    Y AT(6,4):"1: GANZRATIONALE
    FUNKTION       5.GRADES"
210 DISPLAY AT(12,1):"2: FUNKTIO
    N ANDEREN TYPS"
220 DISPLAY AT(17,1):"3: FUNKTIO
    N BEIBEHALTEN" :: DISPLAY AT
    (22,1):"4: WIEDERHOLUNG DER
    WERTE"
230 CALL KEY(0,LL,PP):: IF PP=0
    THEN 230 ELSE IF LL=51 THEN
    290 ELSE IF LL=49 THEN KW=1
    :: GOTO 290 ELSE IF LL=52 TH
    EN 1460
240 DISPLAY AT(4,7)ERASE ALL:"KU
    RVENDISKUSSION" :: CALL HCHA
    R(6,1,61,32)
250 DISPLAY AT(8,1):"DU MUSST MI
    T EINEM >DEF<-   STATEMENT U
    ND DER ENT-      SPRECHENDEN
     ZEILENNUMMER    FOLGENDE DA
    TEN EINGEBEN: "
260 DISPLAY AT(14,2):"ZEILE 370
    FUNKTION: Y(X)=" :: DISPLAY
    AT(16,8):"380 1.ABLEITUNG: Z
    =" :: DISPLAY AT(18,8):"390
    2.ABLEITUNG: G="
270 DISPLAY AT(20,8):"400 3.ABLE
    ITUNG: M="
280 DISPLAY AT(23,1):"NACH EINGA
    BE DAS PROGRAMM   MIT >RUN 2
    90< STARTEN" :: END
290 REM ANFANG
300 GOSUB 1580 :: DISPLAY AT(5,2
    ):"MENUE:"
310 DISPLAY AT(8,1):"1: Y-ACHSEN
    ABSCHNITT";;:"2: NULLSTELLEN
    ";;:"3: EXTREMSTELLEN";;:"4:
     WENDESTELLEN"
320 DISPLAY AT(16,1):"5: GESAMTE
     KURVENDISKUSSION";;:"6: GRA
    PHISCHE DARSTELLUNG";;:"7: W
    ERTETABELLE";;:"8: FLAECHENB
    ERECHNUNG"
330 CALL KEY(0,W,S):: IF S=0 THE
    N 330
340 IF W<49 OR W>56 THEN 330
350 CALL CLEAR :: MAY=0 :: MIY=0
     :: NUL=0 :: WEN=0
360 IF KW=0 THEN 570
370 DEF Y(X)=A*X^5+B*X^4+C*X^3+D
    *X^2+E*X+F
380 DEF Z=5*A*X^4+4*B*X^3+3*C*X^
    2+2*D*X+E
390 DEF G=20*A*X^3+12*B*X^2+6*C*
    X+2*D
400 DEF M=60*A*X^2+24*B*X+6*C
410 FUN$=" ="
420 PRINT "EINGABE DER KOEFFIZIE
    NTEN  EINER GANZRATIONALEN
```

Listing zur »Kurvendiskussion«

```
            FUNKTION MIT DER FORM
    :": :
430 PRINT "Y=A*X^5+B*X^4+C*X^3+D
    *X^2   +E*X+F": :
440 INPUT "A=":A
450 INPUT "B=":B
460 INPUT "C=":C
470 INPUT "D=":D
480 INPUT "E=":E
490 INPUT "F=":F
500 CALL CLEAR
510 IF A<>0 THEN FUN$=STR$(A)&"*
    X^5"
520 IF B>0 THEN FUN$=FUN$&"+"&ST
    R$(B)&"*X^4" ELSE IF B<>0 TH
    EN FUN$=FUN$&STR$(B)&"*X^4"
530 IF C>0 THEN FUN$=FUN$&"+"&ST
    R$(C)&"*X^3" ELSE IF C<>0 TH
    EN FUN$=FUN$&STR$(C)&"*X^3"
540 IF D>0 THEN FUN$=FUN$&"+"&ST
    R$(D)&"*X^2" ELSE IF D<>0 TH
    EN FUN$=FUN$&STR$(D)&"*X^2"
550 IF E>0 THEN FUN$=FUN$&"+"&ST
    R$(E)&"*X" ELSE IF E<>0 THEN
     FUN$=FUN$&STR$(E)&"*X"
560 IF F>0 THEN FUN$=FUN$&"+"&ST
    R$(F)ELSE IF F<>0 THEN FUN$=
    FUN$&STR$(F)
570 PRINT "FUNKTION: Y=";FUN$ ::
     PRINT " "
580 ON (W-48)GOTO 590,610,760,95
    0,590,1210,1590,1100
590 PRINT "Y-ACHSENABSCHNITT: Y=
    ";Y(0):: PRINT :: IF W<>53 T
    HEN 1430
600 REM NULLSTELLEN
610 GOSUB 1450
620 GOTO 640
630 S=S/10 :: TA=SA :: TN=SN
640 FOR X=TA TO TN STEP S :: SN=
    X
650 IF Y(X)=0 OR S=1E-9 THEN 680
     ELSE IF SGN(YN)=0 THEN 660
    ELSE IF SGN(Y(X))<>SGN(YN)TH
    EN 630
```

Zeilen	190 - 330:	Anfangsbilder und Menü
Zeilen	370 - 400:	Stammfunktion und Ableitungen
Zeilen	440 - 490:	Eingabe der Koeffizienten
Zeilen	510 - 570:	Bestimmen der ganzrationalen Funktion
Zeile	580:	Sprungbefehl der gewählten Operationen
Zeile	590:	Y-Achsenabschnitt
Zeilen	600 - 740:	Nullstellenberechnung
Zeilen	750 - 930:	Extremstellenberechnung
Zeilen	940 - 1080:	Wendestellenberechnung
Zeilen	1100 - 1200:	Flächenberechnung
Zeilen	1210 - 1420:	Grafische Darstellung
Zeilen	1460 - 1570:	Wiederholung der Werte
Zeilen	1590 - 1620:	Aufstellen der Wertetabelle

Programmbeschreibung zur »Kurvendiskussion«

A, B, C, D und E sind die Koeffizienten der ganzrationalen Funktion 5. Grades.

J (X)-Nullstellen	Y(X)-Stammfunktion
JM(X)-relative Maxima	Z- 1. Ableitung
JN(X)-relative Minima	G- 2. Ableitung
JW(X)-Wendestellen	M- 3. Ableitung

W-Menünummer der durchzuführenden Operation
FL-Fläche unter dem Graphen

JU-untere Integrationsgrenze JO-obere Integrationsgrenze

Grafische Darstellung:

XO-obere Grenze X-Achse	XU-untere Grenze X-Achse
YO-obere Grenze Y-Achse	YU-untere Grenze Y-Achse

Variablenliste zur »Kurvendiskussion«

```
660 YN=Y(X):: SA=SN :: NEXT X
670 GOTO 710
680 PRINT "NULLSTELLE BEI X=";X
    :: NUL=NUL+1 :: J(NUL)=X
690 CALL SOUND(50,415,2)
700 X=X+.5 :: TA=X :: TN=20 :: K
    =K+1 :: YN=0 :: S=1 :: GOTO
    640
710 IF K<>0 THEN 730
720 PRINT "KEINE NULLSTELLEN"
730 IF W<>53 THEN 1430
740 PRINT
750 REM EXTREMSTELLEN
760 GOSUB 1450
770 GOTO 790
780 S=S/10 :: TA=SA :: TN=SN
790 FOR X=TA TO TN STEP S :: SN=
    X
```

Listing zur »Kurvendiskussion« (Fortsetzung)

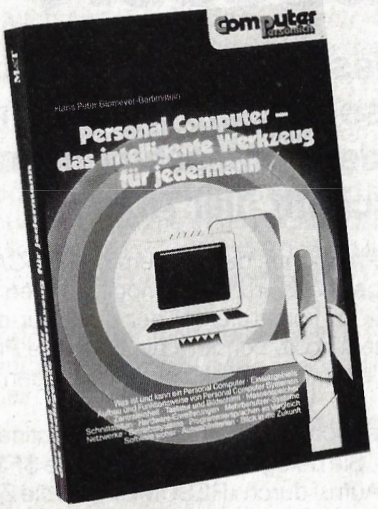
COMPU
LADEN MÜNCHEN
Computer-Betriebs-GmbH
Klenzestraße 38/1 Rgb.
8000 München 2
Tel.: 089/2607002-7
★ EDV-Zubehör ★ ständig Gelegenheiten ★ EDV-Arbeitsplätze ★ Büroeinrichtungen — Ihr Partner — ★ Beratung — Verkauf — Schulungen — eigene Programmierung ★
★ alphaTronic ★ DEC ★ IBM ★ Commodore ★ Sharp ★
Software: — Lohn-/Bau-Lohn — Lager-Faktura — Text-Be-/Verarbeitung — Finanzbuchhaltung — Anlegerberatung (ABS) — Baufinanzierung — Versicherungsmakler — Konzeptionsprogramm — Individualsoftware

```
800 IF Z=0 OR S=1E-8 THEN 830 EL
    SE IF SGN(ZN)=0 THEN 810 ELS
    E IF SGN(Z)<>SGN(ZN)THEN 780
810 ZN=Z :: SA=SN :: NEXT X
820 GOTO 900
830 IF G>=0 THEN 860
840 PRINT "RELATIVES MAXIMUM BEI
            (";X;";";Y(X);")" :
    : MAY=MAY+1 :: JM(MAY)=X
850 GOTO 880
860 IF G=0 THEN 890
870 PRINT "RELATIVES MINIMUM BEI
            (";X;";";Y(X);")" :
    : MIY=MIY+1 :: JN(MIY)=X
880 CALL SOUND(50,415,2)
890 K=K+1 :: X=X+.1 :: TA=X :: T
    N=20 :: ZN=0 :: S=1 :: GOTO
    790
900 IF K<>0 THEN 920
910 PRINT "KEINE EXTREMSTELLEN"
920 IF W<>53 THEN 1430
930 PRINT
940 REM WENDESTELLEN
950 GOSUB 1450
960 GOTO 980
970 S=S/10 :: TA=SA :: TN=SN
980 FOR X=TA TO TN STEP S :: SN=
    X
990 IF G=0 OR S=1E-9 THEN 1020 E
    LSE IF SGN(GN)=0 THEN 1000 E
    LSE IF SGN(G)<>SGN(GN)THEN 9
    70
1000 GN=G :: SA=SN :: NEXT X
1010 GOTO 1060
1020 IF M=0 THEN 1050
1030 PRINT "WENDESTELLE BEI":"("
     ;X;";";Y(X);")" :: WEN=WEN+
     1 :: JW(WEN)=X
1040 CALL SOUND(50,415,2)
1050 K=K+1 :: X=X+.1 :: TA=X ::
     TN=20 :: GN=0 :: S=1 :: GOT
     O 980
1060 IF K<>0 THEN 1090
1070 PRINT "KEINE WENDESTELLEN"
1080 PRINT
1090 GOTO 1430
1100 REM FLAECHENBERECHNUNG
1110 PRINT "FLAECHENBERECHNUNG:"
1120 FL=0 :: PO=0 :: LW=0
1130 INPUT "UNTERE GRENZE XU=":J
     U
1140 INPUT "OBERE GRENZE XO=":JO
1150 RQ=Y(JU)+Y(JO)
1160 FOR Q=JU+2/100 TO JO-2/100
     STEP 2/100 :: PO=PO+Y(Q)::
     NEXT Q :: PO=PO*2
1170 FOR R=JU+1/100 TO JO-1/100
     STEP 2/100 :: LW=LW+Y(R)::
     NEXT R :: LW=LW*4
1180 FL=(RQ+LW+PO)/300
1190 PRINT "FLAECHE UNTER DEM GR
     APHEN        A=";FL:" "
1200 GOTO 1430
1210 REM GRAPHISCHE DARSTELLUNG
1220 CALL CHAR(35,"1010101010101
     010",36,"000000FF",37,"1010
     10FF10101010",38,"103838549
     2921010",39,"300806FF06083"
     )
1230 PRINT "GRAPHISCHE DARSTELLU
     NG"
1240 PRINT " "
1250 INPUT "UNTERE GRENZE X-ACHS
     E XU=":XU
1260 INPUT "OBERE GRENZE X-ACHSE
      XO=":XO
1270 INPUT "UNTERE GRENZE Y-ACHS
     E YU=":YU
1280 INPUT "OBERE GRENZE Y-ACHSE
      YO=":YO
1290 CALL CLEAR
1300 ! BERECHNUNG DES
        KOORDINATENSYSTEMS
1310 KOK=INT((24/(ABS(YO)+ABS(YU
     )))*ABS(YO))
1320 OKO=(32/(ABS(XO)+ABS(XU)))*
     ABS(XU)+1
1330 XAC=0
1340 IF KOK<1 OR KOK>24 THEN 121
     0 ELSE IF OKO<1 OR OKO>32 T
     HEN 1210
1350 CALL HCHAR(KOK,1,36,31):: C
     ALL VCHAR(1,OKO,35,23):: CA
     LL HCHAR(KOK,OKO,37):: CALL
      HCHAR(KOK,31,39):: CALL HC
     HAR(1,OKO,38)
1360 FOR I=XU TO XO STEP (ABS(XO
     )+ABS(XU))/32
1370 FA=KOK-INT((24/(ABS(YU)+ABS
     (YO)))*Y(I)):: XAC=XAC+1
1380 IF XAC>32 THEN 1420
1390 IF FA<=0 OR FA>24 THEN 1410
1400 CALL HCHAR(FA,XAC,42)
1410 NEXT I
1420 CALL KEY(0,K,S):: IF S=0 TH
     EN 1420 ELSE IF K=61 THEN 1
     220
1430 DISPLAY AT(24,5):" MENUE ?
     (J/N)"
1440 CALL KEY(0,K,S):: IF S=0 TH
     EN 1440 ELSE IF K=74 THEN 2
     00 ELSE END
```

```
1450 S=1 :: YN=0 :: ZN=0 :: GN=0
     :: TA=-20.12 :: TN=20 :: K
     =0 :: RETURN
1460 DISPLAY AT(1,7)ERASE ALL:"K
     URVENDISKUSSION" :: CALL HC
     HAR(3,1,61,32):: DISPLAY AT
     (6,1):"1: NULLSTELLEN":;:"2
     : EXTREMSTELLEN"
1470 DISPLAY AT(10,1):"3: WENDES
     TELLEN":;:"4: FLAECHE UNTER
      DEM GRAPHEN"
1480 CALL KEY(0,JK,S):: IF S=0 T
     HEN 1480 ELSE IF JK<49 OR J
     K>52 THEN 1480
1490 GOSUB 1580 :: DISPLAY AT(5,
     1):"FUNKTION: Y=",FUN$
1500 ON JK-48 GOTO 1510,1520,155
     0,1570
1510 FOR I=1 TO NUL :: DISPLAY A
     T(I+7,1):"NULLSTELLE X=";J(
     I):: NEXT I :: GOTO 1430
1520 DISPLAY AT(7,1):"REL. MAXIM
     UM " :: FOR I=1 TO MAY :: D
     ISPLAY AT(I+8,1):"(";JM(I);
     ";";Y(JM(I));")" :: NEXT I
1530 DISPLAY AT(9+MAY,1):"REL. M
     INIMUM " :: FOR I=1 TO MIY
     :: DISPLAY AT(I+9+MAY,1):"(
     ";JN(I);";";Y(JN(I));")"
1540 NEXT I :: GOTO 1430
1550 DISPLAY AT(7,1):"WENDESTELL
     E " :: FOR I=1 TO WEN :: DI
     SPLAY AT(I+7,1):"(";JW(I);"
     ";";Y(JW(I));")"
1560 NEXT I :: GOTO 1430
1570 DISPLAY AT(9,1):"FLAECHE VO
     N XU=";JU;"BIS XO=";JO;":
     FL=";FL :: GOTO 1430
1580 DISPLAY AT(1,7)BEEPERASE AL
     L:"KURVENDISKUSSION" :: CAL
     L HCHAR(3,1,61,32):: RETURN
1590 GOSUB 1580 :: DISPLAY AT(5,
     3):"WERTETABELLE:" :: CALL
     HCHAR(8,3,95,26):: CALL VCH
     AR(9,16,124,15):: QW=0
1600 DISPLAY AT(7,2):"X-WERTE","
     Y-WERTE"
1610 QW=QW+1 :: ACCEPT AT(QW+10,
     2)(NUMERIC,"=")SIZE(10):WER
     T$
1620 IF QW>12 OR WERT$="=" THEN
     1430 ELSE DISPLAY AT(QW+10,
     17):Y(VAL(WERT$)):: GOTO 16
     10
```

Listing zur »Kurvendiskussion« (Schluß).

Personal Computer — das intelligente Werkzeug für jedermann



Dieses Buch ist der Nachfolger des Standardwerkes »Personal Computer-Kompaktrechner im Einsatz«. Es faßt den aktuellen Stand der Personal Computer-Technik zusammen:
Was ist und kann ein Personal Computer · Einsatzgebiete · Aufbau und Funktionsweise von Personal Computer-Systemen · Zentraleinheit · Tastatur und Bildschirm · Massenspeicher · Schnittstellen · Hardware-Erweiterungen · Mehrbenutzer-Systeme · Netzwerke · Betriebssysteme · Programmiersprachen im Vergleich · Software woher · Auswahlkriterien · Blick in die Zukunft

1983, 352 Seiten

Best.-Nr. MT 508
DM 56,—*

*DM 53,— inkl. MwSt. zuzügl. DM 3,— Porto und Versand

computer persönlich
Hans-Pinsel-Str. 2 · 8013 Haar bei München
Telefon 089/4613-220

LESERFORUM

Laut diversen TI-Prospekten und ihrer Heimcomputer-Übersicht in Heft 26/83 besitzt der TI 99/4A zehn frei programmierbare Funktionstasten. Um welche Tasten handelt es sich dabei?

Detlev Bartsch

Bei dieser Angabe handelte es sich um einen Fehler in unserer Übersicht. Alle Tasten können aber mit CALL KEY abgefragt werden. Damit ist eine bedingte Verzweigung in Programmen möglich.

Ich besitze einen TI 99/4a und möchte mit dem TI Impact-Drucker (Epson MX80) eine Hardcopy erstellen. Mit dem in CP 5/83 veröffentlichten Programm wird am Zeilenanfang jeweils eine Null ausgedruckt. Wer kann mir helfen? Der Ausdruck sollte über die RS232-Schnittstelle erfolgen.

Heike Bracke

Ist der Anschluß des MDCR/FS-9600, über den unter der Überschrift »Die Kassette als Massenspeicher« in CP 13/84 berichtet wurde, an einen TI 99/4A über IEC-Bus möglich? Oder über eine andere Schnittstelle?

Heribert Gruber

Ich möchte mit dem TI 99/4A die nächste wahrscheinliche Zahl in einer Serie ausrechnen. Beispiel: Eingabe = 3,6,9,12 — Ausgabe = 15. Geht das? Wenn ja, wie?

Hans Weisser

In CP 13/84 wurden verschiedene Fragen zum TI 99/4A gestellt, von denen ich einige beantworten möchte. Um hochauflösende Grafik ohne allzu großen Speicherplatzverlust zu realisieren, kann man die CHAR-Hexstrings in DATA-Befehle packen — das verringert zwar die Geschwindigkeit, spart aber Speicherplatz. Eine andere Möglichkeit: Einsatz des Apesoft Expanded Grafik Basic, das allerdings eine RAM-Erweiterung beziehungsweise das Mini Memory Modul voraussetzt. Dort hat man 40 neue Grafikbefehle — allerdings kann man dann keine anderen Grafikbefehle des Basic benutzen.

Mit dem Sprachsynthesizer lassen sich auch andere Worte ausgeben als die üblicherweise gespeicherten. Man kann ein Maschinenprogramm in die 32-K-Erweiterung laden und damit seine eigenen Worte abspeichern. Man kann Sprache aber auch mit anderen Modulen, zum Beispiel mit dem Terminal Emulator II, verwalten lassen. Zeitraubend aber nicht sehr wirkungsvoll ist eine andere Methode, die auch nur Extended Basic und den Synthesizer erfordert: Man holt sich die Anfänge bestimmter Wörter mit dem SPGET und kombiniert diese dann miteinander.

Um in Assembler zu programmieren benötigt man mindestens eine 32-KByte-Erweiterung. Besser: Editor-Assembler, Disk-Laufwerk mit Controller und RAM-Erweiterung.

Mit dem GP100A kann man Grafik drucken, indem man den Drucker mit einem OPEN #1:"PIO.LF" öffnet und dann mit PRINT #1:CHR$(8) in den Grafik-Mode umschaltet. Dann kann man die Grafik zeilenweise eingeben. Um eine neue Zeile beginnen zu können, muß der OPEN ohne LF geändert werden. Die Grafik gibt man mittels CHR$-Code ein. Diese müssen zwischen 128 (keine Punkte) und 256 (alle Punkte) liegen. Das probiert man aber am besten selbst aus.

Frank Horn

Ein Tip für alle 99/4A-Besitzer: Hüten Sie sich vor der Tastenkombination »CLEAR« und »M«, also dem gleichzeitigen Drücken der drei Tasten »FCTN«, »%« und »M«, denn sie hat dieselbe Wirkung wie »QUIT«. Wenn Sie allerdings die Möglichkeit haben, auf einzelne Speicherzellen zuzugreifen, zum Beispiel mit CALL LOAD, können Sie beides abschalten, indem Sie an die Speicherzelle Hex 83C2/Dec-31806 den Wert 16 schreiben.

Christian Weichmann

Wer kennt den VC 20?

Ich bekam ein Forth-Modul für den VC 20 geschenkt — leider ohne Anleitung. Das Modul stammt von Audio Genic. Wo kann ich eine Anleitung kaufen oder ein für mich geeignetes Buch über Forth?

Ulrich Seeger

Ich will die Brother EP20 als Drucker für den VC 20 benutzen. Wer hat Unterlagen für den Selbstbau eines Interfaces, bietet ein fertiges Interface an oder kann die Brother umrüsten?

Detlev Junior

Gibt es fertige Programme zum Drucken von Briefköpfen mit VC 20 und Drucker MPS801?

Klaus Kohler

Ich habe von einer Firma gehört, die den VC 20 zum C 64 umrüstet. Wer hat Erfahrungen mit einer solchen Umrüstung?

Dieter Ludwig

Wer kennt den 64?

Ich habe ein Schuhgeschäft und arbeite mit 8stelligen Artikelnummern. Ich suche ein für mich geeignetes Lagerverwaltungsprogramm, das auf Commodore 64 mit Floppylaufwerk laufen soll. Ich muß nach folgenden Kriterien sortieren können: Schuhgattung, Farbe, Saison, Lieferant. Das Programm sollte es außerdem erlauben, eine Verkäuferstatistik zu führen, Prämienberechnungen durchzuführen und Etiketten zu drucken. Wer kennt geeignete Software?

Hans Bardenhagen

Wer liefert ein Interface, um den C 64 an das Floppylaufwerk 8050 von Commodore anzuschließen?

Thomas Lopatic

Commodore liefert für den 64 ein IEC-Bus-Modul. Über dieses Modul kann die 8050 angeschlossen werden.

Ich suche für unseren Verein ein Mitgliederverwaltungsprogramm, das auf dem Commodore 64 mit Floppy 1541 und Drucker 1526 läuft. Wer hat Erfahrungen mit einem solchen Programm? Wer kennt Lieferanten?

Horst Kessler

Wer hat Erfahrungen mit Bar-Code-Lesern in Verbindung mit dem Commodore 64 und verfügt über entsprechende Software?

Markus Puetz

Ich habe einen C 64. Am Joy-Port 2 wird der Feuerknopf nicht mehr abgefragt. Eine mechanische Beschädigung ist ausgeschlossen. Da bei einem Bekannten der gleiche Mangel auftrat und sein C 64 bereits 4 Wochen in Reparatur ist, wollte ich fragen, ob andere Leser dieses Problem kennen oder helfen können.

Peter Fürle

Microdrive für Oric 1?

Wo bleibt eigentlich das Microdrive für den Oric 1? Die Firma kündigte doch so etwas an, als der Computer herauskam. Paßt das Microdrive von Sinclair irgendwie?

Frank Werner

Nachdem Oric jetzt den Atmos herausgebracht hat und dazu auch ein Disketten-Laufwerk, dürfte es wohl für den Oric 1 kein eigenes Microdrive mehr geben. Es ist kaum anzunehmen, daß das Microdrive von Sinclair sich sinnvoll anschließen läßt.

Lösung für Ulysses?

Wo ist bei dem Abenteuerspiel Ulysses and the golden Fleece das Zauberschwert zu finden, mit dem die Skelette auf der großen Insel zerstört werden, und wie und wann ist die CHEST zu öffnen?

Alger Blömeke

BetaBasic auf Kassette?

Ich besitze das BetaBasic für den Spectrum. Da ich mir das Microdrive hinzugekauft habe, möchte ich nunmehr BB auf eine Cartridge schreiben. Können Sie mir den notwendigen Befehlsablauf nennen?

Bernd Kalusa

Fragen Sie doch

Selbst bei sorgfältiger Lektüre von Handbüchern und Programmbeschreibungen bleiben beim Anwender immer wieder Fragen offen. Viel mehr Fragen ergeben sich bei Computer-Interessenten, die noch keine festen Kontakte zu Händlern, Herstellern oder Computerclubs haben. Sie können der CP-Redaktion Ihre Fragen schreiben oder Probleme schildern (am einfachsten auf der vorn beigehefteten Karte »Lesermeinung«). Wir veranlassen, daß sie von einem Fachmann beantwortet werden. Allgemein interessierende Fragen und Antworten werden veröffentlicht, die übrigen brieflich beantwortet.

Stromkosten auf einen Blick

Die Strom- oder Gasrechnung kommt meist nur einmal im Jahr ins Haus. Mit diesem Programm, das in TI-Basic geschrieben ist, kann man diese Nebenkosten im Haushalt jederzeit kontrollieren, wenn man den Zählerstand abgelesen hat.

Das Programm »Stadtwerke-Abrechnung« dient zur schnellen Ermittlung des Energieverbrauchs und der Kosten von Strom und Gas im eigenen Haushalt.

Nach dem Starten des Programms mit »RUN« erscheint das Auswahlmenü:

Strom = 1
Gas = 2

Es folgt die Abfrage des Eingabedatums. Das Programm sieht Berechnungen zwischen den Jahren 1901 und 2099 vor.

Es beginnt die Abfrage der Eingabedaten:

— täglicher Meßpreis (einige Bundesländer haben einen jährlichen Meßpreis; in dem Fall muß durch 365 geteilt werden).
— Arbeitspreis für Strom/Gaspreis (als DM-Betrag eingeben)
— Ausgleichsabgabe (nur bei Strom)
— Mehrwertsteuersatz

Es folgt die Überprüfung der Eingabedaten mit Korrekturmöglichkeit. Das Programm beginnt mit der Berechnung, zum Beispiel für Strom, und gibt die erste Darstellung auf den Bildschirm aus:

— Abrechnungszeitraum
— Grundmeßpreis
— effektiver Verbrauch
— Nettobetrag
— Ausgleichsabgabe
— Mehrwertsteuer

= Rechnungsbetrag

Anschließend erscheint die Frage: Sollen weitere Berechnungen (zum Beispiel für Strom) durchgeführt werden? Wenn »ja« (zum Beispiel für Gas) beginnt die Berechnung von neuem. Es folgt die Darstellung auf dem Bildschirm.

Werden keine weiteren Berechnungen mehr gewünscht, folgt die Endberechnung. Die Endberechnung führt noch mal alle Einzelposten für Strom und Gas auf und gibt die Gesamtbeträge aus. Die Gesamtbeträge setzen sich aus allen Rechnungsbeträgen der bisher gerechneten Durchgänge zusammen. Die Gesamtsumme ist der endgültige Betrag.

(Erwin Bath)

```
10 REM
  ********************
  *    PROGRAMM BY   *
  * E.BUTH & S.SCH.*
  ********************
15 CALL CLEAR
20 N1=0
25 N2=0
30 N3=0
35 A$="FEHLER-DATUMEINGABE I
NKOREKT"
40 CALL CHAR(134,"0000FF00FF
")
45 CALL CHAR(133,"3C4299A1A1
99423C")
50 CALL CHAR(128,"000000FFFF
")
55 CALL CHAR(129,"000000FF")
60 CALL CHAR(130,"FFFFFFFFFF
FFFFFF")
65 CALL CHAR(131,"0101010101
010101")
70 CALL CHAR(132,"8080808080
808080")
75 CALL CHAR(104,"00404058644
444444")
80 CALL CHAR(110,"0070103010
70")
85 CALL CHAR(109,"0000000014
2A2A2A")
90 CALL SCREEN(8)
95 CALL COLOR(1,16,1)
100 CALL COLOR(2,16,1)
105 CALL COLOR(3,16,1)
110 CALL COLOR(4,16,1)
115 CALL COLOR(5,16,1)
120 CALL COLOR(6,16,1)
125 CALL COLOR(7,16,1)
130 CALL COLOR(8,16,1)
135 CALL COLOR(10,16,1)
140 CALL COLOR(13,16,1)
145 REM ***EINLEITUNG***
150 PRINT TAB(2);"**********
**************":TAB(2);"*";
TAB(26);"*"
155 PRINT TAB(2);"* STADTWER
KE/ABRECHNUNG *":TAB(2);"*";
TAB(26);"*":TAB(2);"********
******************": : : : :
```

Listing »Stadtwerke-Abrechnung«

```
: : :
160 PRINT TAB(12);CHR$(133);
CHR$(49);CHR$(57);CHR$(56);C
HR$(52)
165 REM**********************
      *PROGRAMM BY        *
      *E.BUTH  &  S.SCH.  *
      **********************
170 FOR I=1 TO 500
175 NEXT I
180 CALL CLEAR
185 CALL SCREEN(8)
190 PRINT TAB(6);"BITTE WAEH
LEN SIE": : : :
195 PRINT TAB(10);"1=> STROM
":TAB(10);"2=> GAS": : : : :
 : : :
200 CALL KEY(0,K,S)
205 IF S=0 THEN 200
210 IF K<49 THEN 200
215 IF K>50 THEN 200
220 REM ***HAUPTPROGRAMM***
225 GOTO 245
230 PRINT : : : : : :A$
235 FOR I=1 TO 400
240 NEXT I
245 CALL CLEAR
250 CALL SCREEN(8)
255 PRINT TAB(6);"EINGABEDAT
UM VON:": : : : :
260 INPUT "TAG   :":T
265 INPUT "MONAT :":M
270 INPUT "JAHR  :":J
275 IF T>31 THEN 230
280 IF M>12 THEN 230
285 IF J>2099 THEN 230
290 IF J<1901 THEN 230
295 GOSUB 965
300 N1=N
305 GOTO 330
310 PRINT : : : : : :A$
315 FOR I=1 TO 400
320 NEXT I
325 CALL CLEAR
330 PRINT : : :TAB(6);"EINGA
BEDATUM BIS:": : : : :
335 INPUT "TAG   :":T
340 INPUT "MONAT :":M
```

Listing »Stadtwerke-Abrechnung« (Fortsetzung)

Aktuell für Ihren PC:

SPEEDY 100-80
80 cps Matrixdrucker A4, Einzelblatt und Traktor, Grafik, EPSON-kompatibel Centronics-Schnittstelle
Knüllerpreis 799-

JUKI 6100
Profi-Typenraddrucker A4 quer, 22 cps, 2K Puffer, WordStar kompatibel, Centronics-Schnittstelle
Knüllerpreis 1899-

TEAC FD 55-A
5.25"-Diskettenlaufwerk für APPLE, 143 KB, kpl. mit Kabel und Gehäuse 699-
Controller-Karte 150

VERBATIM
5.25"-Disketten DATALIFE SS/DD, Lochverstärkung, softsektoriert, 48 tpi, mit Reinigungsset
100 Stück 599-

Barpreise inkl. MwSt. Abholung oder NN-Versand.

cvb computer
Repsoldstraße 49, 2000 Hamburg 1
Tel. 040-230685
Apple · NCR · Epson · NEC · Sanyo · Verbatim · Taxan · 3M · M+T
Info anfordern!

HELPWARE!

C BASIC Anwenderhandbuch
von Adam Osborne, Gordon Eubanks & Martin McNiff *224 Seiten, DM 39,80*

BASIC für den IBM Personal Computer
von Herbert Peckmann *348 Seiten, DM 35,–*

Praktische BASIC-Programme für den IBM Personal Computer
von Lon Poole (Hrsg.) *172 Seiten, DM 29,80*

IBM-PC Compact
Diese Faltkarte bietet alle BASIC-Befehle, Funktionen und Variablen auf einen Blick.
21 x 9,5 cm, DM 9,80

McGraw Hill
Hamburg

Fordern Sie unseren Gesamtprospekt an! Coupon ausschneiden und einsenden an:

McGraw-Hill Book Co. GmbH
Lademannbogen 136
2000 Hamburg 63

Bitte senden Sie mir Ihren kostenlosen Gesamtprospekt COMPUTER-LITERATUR!
Name .....
Anschrift .....

```
345 INPUT "JAHR  :":J
350 IF T>31 THEN 310
355 IF M>12 THEN 310
360 IF J>2099 THEN 310
365 IF J<1901 THEN 310
370 GOSUB 965
375 N2=N
380 N3=N2-N1
385 CALL CLEAR
390 PRINT "EINGABE DER DATEN
:": : : : : :
395 INPUT "ZAEHLERSTAND
ALT:":Z1
400 INPUT "
NEU:":Z2
405 PRINT
410 INPUT "TGL.MESSPREIS DM/
TAG:":TGM
415 AUS=0
420 IF K=50 THEN 440
425 INPUT "ARBEITSPREIS  DM/
KWh:":P
430 INPUT "AUSGL.ABGABE  IN
 % :":AUS
435 GOTO 445
440 INPUT "GASPREIS      DM/
m³ :":P
445 INPUT "STEUERSATZ    IN
 % :":ST
450 CALL CLEAR
455 PRINT TAB(4);"EINGABEDAT
EN RICHTIG ?": : : :
460 CALL HCHAR(22,1,128,32)
465 PRINT "ZAELERST.  ALT :"
;Z1:TAB(11);" NEU :";Z2: : :
470 PRINT "TGL. MESSPREIS :"
;TGM: :
475 PRINT "ARBEITSPREIS   :"
;P: :
480 IF K=50 THEN 490
485 PRINT "AUSGL.ABGABE % :"
;AUS: :
490 PRINT "STEUERSATZ   % :"
;ST: : : :
495 CALL HCHAR(22,1,128,32)
500 PRINT "RICHTIG  (JA1,NEI
N2)"
505 CALL KEY(0,K1,S1)
510 IF S1=0 THEN 505
515 IF K1=50 THEN 385
520 IF K1<>49 THEN 505
525 V=Z2-Z1
530 GOSUB 1005
535 IF K=50 THEN 565
540 NB0=NB0+NB
545 MWST0=MWST0+MWST
550 V0=V0+V
555 GB=GB+RB
560 GOTO 585
565 NB1=NB1+NB
570 MWST1=MWST1+MWST
575 V1=V1+V
580 GB1=GB1+RB
585 CALL CLEAR
590 GS=GB+GB1
595 REM **AUSGABE BERECHNUNG
600 CALL SCREEN(2)
605 PRINT TAB(5);"AUSGABE: B
ERECHNUNG!": :
610 PRINT "ZEITRAUM FUER    :
";N3;TAB(25);"TAGE": :
615 IF K=50 THEN 630
620 PRINT "VERBRAUCH        :
";V;TAB(26);"KWh": :
625 GOTO 635
630 PRINT "VERBRAUCH        :
";V;TAB(27);"m³": :
635 PRINT "GRUNDMESSPREIS   :
";GM;TAB(27);"DM": :
640 PRINT "EFF. VERBRAUCH   :
";EV;TAB(27);"DM": :
645 PRINT "NETTOBETRAG      :
";NB;TAB(27);"DM": :
650 PRINT "AUSGL.ABGABE     :
";AGL;TAB(27);"DM": :
655 PRINT "MEHRWERTSTEUER   :
";MWST;TAB(27);"DM": :
660 PRINT "RECHNUNGSBETRAG  :
";RB;TAB(27);"DM": :
665 CALL HCHAR(7,3,128,28)
670 CALL HCHAR(6,3,130,3)
675 CALL HCHAR(11,3,129,28)
680 CALL HCHAR(15,3,129,16)
685 CALL HCHAR(15,20,129,11)
690 CALL HCHAR(6,27,130,4)
695 CALL HCHAR(21,20,129,11)
700 CALL HCHAR(23,3,128,28)
```

Listing »Stadtwerke-Abrechnung« (Fortsetzung)

```
705 CALL VCHAR(6,2,131,18)
710 CALL VCHAR(6,31,132,18)
715 IF K=50 THEN 730
720 PRINT :"WEITERE STROM-BE
RECHNUNG?"
725 GOTO 735
730 PRINT :"WEITERE GAS-BERE
CHNUNG?"
735 PRINT "(JA1,NEIN2)"
740 CALL KEY(0,K1,S1)
745 IF S1=0 THEN 740
750 IF K1=49 THEN 245
755 IF K1<>50 THEN 740
760 CALL CLEAR
765 PRINT TAB(4);"WEITERE BE
RECHNUNGEN":TAB(4);"FUER STR
OM ODER GAS?": : : : :TAB(9)
;"(JA1,NEIN2)": : : : : : :
:
770 CALL KEY(0,K1,S1)
775 IF S1=0 THEN 770
780 IF K1=49 THEN 180
785 IF K1<>50 THEN 770
790 REM ***AUSGABE GES.***
795 CALL CLEAR
800 PRINT TAB(4);"ENDBERECHN
UNG-STROM/GAS": :
805 PRINT "STROMVERBRAUCH :"
;V0;TAB(25);"KWh": :
810 PRINT "NETTOBETRAG     :"
;NB0;TAB(26);"DM": :
815 PRINT "MEHRWERTSTEUER :"
;MWST0;TAB(26);"DM": :
820 PRINT "GESAMTBETRAG    :"
;GB;TAB(26);"DM": :
825 PRINT "GASVERBRAUCH    :"
;V1;TAB(26);"mn": :
830 PRINT "NETTOBETRAG     :"
;NB1;TAB(26);"DM": :
835 PRINT "MEHRWERTSTEUER :"
;MWST1;TAB(26);"DM": :
840 PRINT "GESAMTBETRAG    :"
;GB1;TAB(26);"DM": :
845 PRINT "GESAMTSUMME     :"
;GS;TAB(26);"DM": : :
850 CALL HCHAR(3,3,130,2)
855 CALL HCHAR(3,30,130)
860 CALL HCHAR(4,3,128,28)
865 CALL HCHAR(10,19,129,11)
870 CALL HCHAR(12,3,128,28)
875 CALL HCHAR(18,19,129,11)
880 CALL VCHAR(3,2,131,21)
885 CALL VCHAR(3,31,132,21)
890 CALL HCHAR(20,3,129,27)
895 CALL HCHAR(22,19,134,11)
900 CALL HCHAR(23,3,128,28)
905 PRINT "WEITER: TASTE DRU
ECKEN!"
910 CALL KEY(0,K1,S1)
915 IF S1=0 THEN 910
920 CALL CLEAR
925 PRINT TAB(5);"WEITERE BE
RECHNUNGEN":TAB(9);"HINZUFUE
GEN?": : : : : : : : : :
930 PRINT TAB(9);"(JA1,NEIN2
)": :TAB(3);"MIT´NEIN´ENDE D
ES PROGRAMMS!"
935 CALL KEY(0,K1,S1)
940 IF S1=0 THEN 935
945 IF K1=49 THEN 180
950 IF K1<>50 THEN 935
955 CALL CLEAR
960 END
965 REM ***TAGE-BERECHNUNG**
**************************

970 N=0
975 IF M>=3 THEN 990
980 J=J-1
985 M=M+12
990 M=M+1
995 N=INT(J*365.25)+INT(M*30
.6)+T
1000 RETURN
1005 REM *POSTENBERECHNUNG**

1010 DEF RDN(X)=INT(X*100+0.
5)*0.01
1015 GM=RDN(N3*TGM)
1020 AGL1=RDN(AGL1+MWSA)
1025 EV=RDN(V*P)
1030 NB=GM+EV
1035 AGL=RDN(NB*AUS/100)
1040 MWST=RDN((NB+AGL)*ST/10
0)
1045 RB=MWST+NB+AGL
1050 RETURN
```

Listing »Stadtwerke-Abrechnung« (Schluß)

LESERFORUM

ProDOS-Tips

Zu Ihrem Artikel über ProDOS Teil I habe ich einige Anmerkungen. Diese stützen sich teilweise auf eigene Versuche und auf einen Artikel »INTRO TO PRODOS« von S. Mossberg in NIBBLE vom März und April 1984. ProDOS prüft die Titelmeldung Apple u. a. sowie den Cursoraufbau. Es wird somit auch in Nachbauten nicht ohne weiteres lauffähig sein (auch bei 64 KByte). Integer-Basic kann geladen werden und ist dann auch arbeitsfähig. Da es aber keinerlei DOS-Kommandos überträgt, ist es wohl kaum nutzbar. Vielleicht hat einer der Leser bereits eine Möglichkeit gefunden, ein benutzbares INTBASIC.SYSTEM zu erstellen. Das RUN-Kommando läuft nicht wie gewohnt. Bitte geben Sie ein: RUN 20 DATEI (nicht mit der Basiceingabe RUN20 verwechseln). Man kann also nun auch bei Filenamenangabe die Startzeile angeben. CREATE kann mit Hilfe des Typparameters jedes File erzeugen. Als Standardwert wird TDIR angenommen. Dieses Directory beschreiben kann man durch SAVE/VOLUME/SUBDIRECTORY /FILE; den Katalog sieht man mit CATALOG/VOLUME/SUBDIRECTORY.

Im Catalog sieht man unter SUBTYPE bei Binärprogrammen die Startadresse. Bei Dateien sieht man die Recordlänge, die bei sequentiellen Dateien als R=O gezeigt wird. Bei DOS 3.3 gab es keine Möglichkeit, dies zu erkennen. Mit APEND können Daten an verschiedene Dateitypen angehängt werden und nicht nur an Textdateien. Mit BLOAD können auch Teile einer Binärdatei geladen werden. Wenn ein File durch ein kürzeres mit gleichem Namen überschrieben wird, merkt ProDOS das und gibt auch nur die kleinere Blockanzahl an. Bei DOS 3.3 ist das keineswegs so (vorher Delete eingeben). Auch neue Befehle selbst zu generieren ist viel leichter möglich geworden. Die Adresse wird unter BE06 bis BE08 abgelegt und die Parameter bei BE50 bis BE56. Es sollen auch noch EXERCISE, TOOLKIT und BUGBYTER angeboten werden. Wenn jemand diese bereits kennt, wäre ich an einem Ideenaustausch sehr interessiert.

Ralph Hendrix

99/4A-Tip

In CP 13/84 wurden verschiedene Fragen zum TI 99/4A gestellt, von denen ich hier zwei beantworten möchte. Eine Absolutadresse von GPLLNK bei Extended-Basic existiert nicht, da es sich bei GPLLNK um eine der Hilfsroutinen handelt, die aus dem G-ROM ausgeladen werden; beim Extended-Basic fehlt diese Routine.

Einige Hinweise zu den DSRs sind in »TI 99/4A Console and Peripheral Expansion System Technical Data«, die Texas Instruments einmal gegen einen Unkostenbeitrag von 25,— DM verschickte, enthalten. Das Softwareinterface zwischen Basic und Assembler ist wie folgt aufgebaut: Ausführung des Assembler-Programms erfolgt über den GPL-Befehl XML (nicht zu verwechseln mit XMLLNK). Werden mit CALL LINK Variablen übergeben, so befindet sich auf 8312H die Anzahl der Variablen und ab 8300H jeweils ein Byte pro Variable, in dem ein Kennwert für den Typ steht. Die Pointer zu den Werten der Variablen liegen im VDP oberhalb des Value Stack, aber dies ist nicht sehr interessant, da ja die Hilfsroutinen NUMASG, STRASG NUMREF UND STRREF zur Verfügung stehen, die die Werte der Variablen in FAC holen beziehungsweise den Variablen Werte zuordnen. Die Frage nach dem Kassettenaufzeichnungsverfahren hier zu beantworten, würde zu weit führen. Es soll hier der Hinweis genügen, daß die Routinen im Betriebssystem ab 1346H (Kassette Write), 142EH (Kassette Read) und 1426H (Kassette verify) zu finden sind (da es mehrere Betriebssysteme beim TI 99 gibt: auf OCF4 bis OCF8H stehen die Pointer zu den Routinen). Um diese Routinen direkt von Assembler-Programmen ohne GPLLNK aufrufen zu können, muß aber vorher noch einiges im CPU-RAM gesetzt werden.

Heiner Martin

Star-Tip

In CP 11/84 berichtet Andreas Österle über Probleme mit dem Drucker Star DP-515. Herr Österle muß eine amerikanische Ausführung des Druckers erhalten haben, da sich mein Drucker (»STAR DP-515«) beim Einschalten automatisch auf den deutschen Zeichensatz umstellt.

Es ist allerdings möglich, mit einem Modell I von Tandy auf andere Zeichensätze umzustellen. Hierzu ist ein Treiberprogramm (Basic-Programm) an den Drucker zu senden.

Der Weg, den Herr Österle beschreitet, ist prinzipiell richtig, nur muß er eben das »Escape R 2« in der einzig möglichen Form senden. Die lautet:

LPRINTCHR$(27)"RR":CHR$(x), wobei X die Zahl für den gewünschten Zeichensatz darstellt.

Die Umstellung funktioniert bei meinem Drucker einwandfrei. Die Ausgabe von Blockgrafik ist aus dem Betriebssystem »NEWDOS 80« möglich, wenn die Option für die Hardcopy (»JKL«) mit "Y" eingesetzt worden ist. Näheres ist im Handbuch für das DOS zu finden.

Allerdings druckt der DP-515 vom deutschen Zeichensatz aus keine Einzelpunkte, sondern nur Sonderzeichen. Mit einem Epson MX80 funktioniert die Ausgabe von Blockgrafik beziehungsweise Hardcopy einwandfrei. Ein anderes Verfahren zur Ausgabe müßte den Inhalt des Video-RAMs »PEEKEN« und dann über die Option »Bit-Image« — Grafik ausgeben.

Reinhold Karn

Kummer mit Vicfile

In CP 14/84 fragte Horst Girschick nach Vicfile. Ich arbeite schon seit längerem erfolgreich mit diesem Programm, muß allerdings bestätigen, daß die Handbücher sehr schlecht sind (das gilt auch für Vicwriter, Simplicalc und Minimemb). Leider habe ich auch noch keine Zeitschrift gefunden, die sich mit diesen Programmen auseinandergesetzt hat. Mein Rezept: Ich schreibe die Quintessenz solcher Handbücher auf DIN A5-Karteikarten, so daß bei Problemen ein schneller Zugriff auf die jeweils benötigte Information möglich ist.

Harald Metz

So schützt man die Spectrum-Tastatur

Um beim Spectrum die Tastaturbeschriftung vor dem »Verschwinden« zu sichern, habe ich die gesamte Frontfläche mit Plastik 70 von Kontakt-Chemie eingesprüht. Man kann den Lack auch mit einem feinen Pinsel auftragen.

Dietmar Belloff

So läuft Genie III langsamer

In CP 15/84 klagte H.C. von Bothmer, daß seine TRS-80-Spiele auf dem Genie III viel zu schnell laufen. Pin 15 der CPU-Platine muß mit Masse verbunden werden — dann läuft das Gerät mit 1,73 MHz. Tonausgabe ist nur durch Programmänderung möglich.

R. Gessenhardt (tcs)

Hier sind Clubs

Für Benutzer von Apple II und Alphatronic aus der Osnabrücker Gegend haben wir einen Computerclub gegründet. Kontaktadresse: Ralf Clasen, Grüner Brink 22, 4513 Belm.

Ralf Clasen

Mit dem TI 99/4A befaßt sich der Compy Computer Club, der derzeit 130 Mitglieder hat. Der Mitgliedsbeitrag beträgt 2 Mark pro Monat (Jugendliche: 1 Mark). Es gibt ein zweimonatlich erscheinendes Clubmagazin und eine Bibliothek mit zirka 250 Tauschprogrammen. Kontaktadresse: Buchenweg 7, 4178 Kevelaer 5.

Erfahrungs- und Softwaretausch sowie regelmäßige Treffen sind Ziele der ersten Bremer-TI 99-Usergroup. Kontaktadresse: Gemeinschaftszentrum Obervieland, Alfred-Faust-Str. 4, 2800 Bremen 61.

Gerhard Hollnagel

Wollen Sie antworten?

Wir veröffentlichen auf dieser Seite auch Fragen, die sich nicht ohne weiteres anhand eines guten Archivs oder aufgrund der Sachkunde eines Herstellers beziehungsweise Programmierers beantworten lassen. Das ist vor allem der Fall, wenn es um bestimmte Erfahrungen geht oder um die Suche nach speziellen Programmen. Wenn Sie eine Antwort auf eine hier veröffentlichte Frage wissen — oder eine andere, bessere Antwort als die hier gelesene, dann schreiben Sie uns doch. Antworten publizieren wir in einer der nächsten Ausgaben. Bei Bedarf stellen wir auch den Kontakt zwischen Lesern her.

TIPS&TRICKS

IBM-PC — Komma auf der Zehnertastatur

Es hat mich immer geärgert, daß ich bei Programmen wie Multiplan deutsch oder Open Access kein Komma auf dem numerischen Tastatur-Block habe. Darum habe ich zu einem einfachen Hilfsmittel gegriffen, das jeder mit den Standard-Programmen des DOS zur Verfügung hat.

Kopieren Sie das Programm KEYBGR.COM auf die Ladediskette für das Programmsystem, das Sie benutzen wollen.

Legen Sie die nationale DOS-Diskette ein. Rufen Sie das Debug-Programm auf:

A)DEBUG

Tauschen Sie die DOS-Diskette gegen die erstellte Ladediskette aus und geben ein:

Für DOS 1.1	Für DOS 2.0
—N KEYBGR.COM	—N KEYBGR.COM
—L	—L
—E 692 2C	—E 696 2C
—W	—W
—Q	—Q

Danach sollten Sie einen Systemstart mit der geänderten Diskette ausführen. Wenn KEYBGR über die Datei AUTOEXEC.BAT geladen wird, können Sie den Punkt als Komma benutzen.

(B. Trede)

TI 99/4A — Mehrfachbenutzung von CALL-Routinen

Mir ist in vielen Listings aufgefallen, daß viele Anwender die mehrfache Benutzung von einigen CALL-Routinen nicht kennen. So ersetzt zum Beispiel CALL COLOR (1,3,4,2,6,13,3, 14,15,4,16,2) die einzelnen Befehle

CALL COLOR(1,3,4)
CALL COLOR(2,6,13)
CALL COLOR(3,14,15)
CALL COLOR(4,16,2).

Für die Besitzer der 32 K-Erweiterung ist es jetzt möglich, durch den Befehl CALL INIT: :CALL LOAD(-31931,0) den PROTECTED-Schutz von ihren Programmen zu entfernen.

(A. Bey)

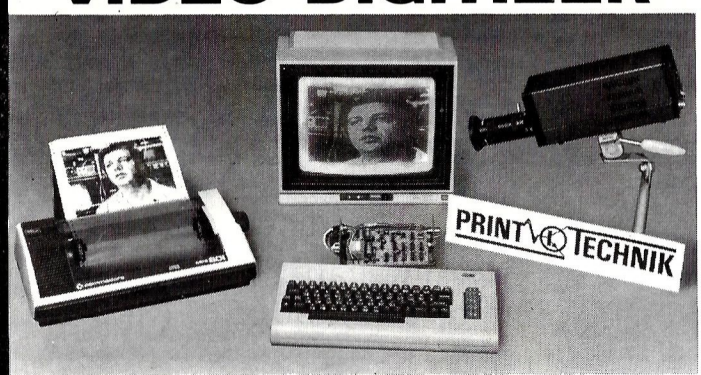
PRINT TECHNIK

IHR PARTNER FÜR 1 A-BAUTEILE. LEITERPLATTEN UND COMPUTER

AUTORISIERTER COMMODORE GROSSHÄNDLER FÜR VC 20 UND C 64
OFFIZIELLER PHILIPS BAUELEMENTE DISTRIBUTOR SIEMENS FACHBERATER

1060 WIEN, STUMPERGASSE 34 TEL.0222/57 34 23 TELEX. 112996

VIDEO-DIGITIZER



Ein neuer Print-Technik-Bausatz, mit dem man jedes Videosignal in zirka 4 Sekunden in den Speicher eines Commodore 64 einlesen kann. Ein Graphikausdruck, Abspeichern auf Floppy sowie Weiterverarbeiten des Bildes ist möglich ..... **Bausatz** DM 458.–

Farbausdruck möglich mit CANON A-1210

Steckmodul Hase ladet Ihre Cassettenprogramme mit der Schnelligkeit einer Floppy-Disc ..... DM 198.–

Steckmodul HELP, unser Verkaufsschlager, 22 Programmierhilfebefehle, 14 Befehle für Maschinensprachprogrammierung, 9 DOS-Befehle (zur vereinfachten Floppybedienung) ..... DM 148.–

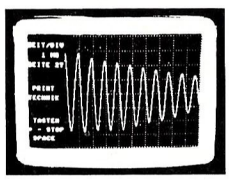
HELP PLUS für **VC 20** und **C 64,** zusätzlich mit Kompaktor und 2-Paß-Assembler ..... DM 248.–

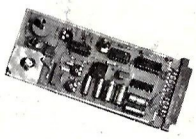
Speicheroszilloskop-Interface für Commodore 64

Mit unserer neuesten Bausatzentwicklung wird es nun endlich möglich, extrem langsame wie auch schnelle Abläufe preiswert zu speichern und oszillographisch darzustellen. Steckmodul mit AD-Wandler, Hardware-Triggerung, Y-Verstärker (10 V, 1 V, 100 mV) pro Rastereinheit, dazugehörige komfortable Software mit kalibrierter Zeitbasis und Raster.

Darstellung auf 1-95 Bildseiten

Ausplotten auf VC 1520 möglich ..... **Bausatz** DM 298.–





NEU: Intelligentes Centronix-Interface

mit Kabel und Stecker zum Anschluß an den seriellen IEEE-Bus des C 64 oder VC 20. Betriebsbereit ..... DM 228.–

Das NEUE Print-Technik-„CW/RTTY 84"-Modul für C 64, das wirklich keine Wünsche offen läßt:

zum Senden und Empfangen sämtlicher Morse- und Funkfernschreibsignale, mit allen nur erdenklichen Möglichkeiten, wie Bildschirmtrennung, Druckerausgabe, Speichermöglichkeiten bis 30.000 Zeichen usw., usw., usw. ..... DM 248.–

Sende- und Empfangskonverter für RTTY, dazupassend mit LED-Anzeige ..... Bausatz DM 158.–

E-Prom-Programmierer, ermöglicht mit Ihrem C 64 sämtliche gängigen E-Proms zu programmieren ..... Bausatz DM 228.–

IEEE-Interface, zum Beispiel für CBM-Peripherie ..... Bausatz DM 198.–

LIGHT-PEN-Bausatz für VC 20 sowie für C 64 zum Registrieren und Positionieren am Bildschirm ..... Bausatz DM 48.–

TYPENRAD-SCHREIBMASCHINE brother CE-50 ALS DRUCKER UND EINGABEEINHEIT FÜR COMMODORE 64 UM NUR .. DM 1198.–

FÜR UNSERE KUNDEN IN DER SCHWEIZ:
2542 PIETERLEN BAHNHOFSTR.2
TEL. 032/87.24.29

MICROTRON COMPUTERPRODUKTE

FÜR UNSERE KUNDEN IN DER BRD:
Fa. COMPUTERPERIPHERIEN
Pitt Joern Brockner 8000München40
HeidelbergerSTR.6 Tel.089/368197

GEGEN EINSENDUNG VON DM5.– ERHALTEN SIE UNSEREN BAUSATZ-KATALOG +BAUSATZBESCHREIBUNGEN

TÄGLICHER NACHNAHMEVERSAND - WIR SIND FÜR SIE DA . MONTAG - FREITAG. 9 -18 UHR

Schotten-muster vom Monitor

Auch beim Weben mit der Hand kann ein Computer helfen. Gerade Anfängern fällt es häufig schwer, Farben für die Kett- und Schußfäden auszuwählen und die Übersicht zu behalten. Ein Listing für den TI 99/4A mit Extended-Basic-Modul macht die Weberei zum Kinderspiel.

Mit diesem Computerprogramm kann man Streifen- und Karomuster auf dem Bildschirm entwerfen. Es wird dabei die einfachste Gewebebindung, nämlich die Leinenbindung, simuliert. Das heißt, beim Einziehen des Schußfadens bleibt jeder zweite Kettfaden unten und damit unsichtbar.

Nach dem Titelbild zeigt das Programm zuerst eine Tafel der 15 verfügbaren Farben an. Davon kann man für den Entwurf bis zu sechs vorauswählen. Will man weniger Farben verwenden, kann man mit der Eingabe »0« diese Prozedur abkürzen. Das nächste Bild zeigt die ausgewählten Farben und ordnet sie den Ziffertasten 1 bis 6 zu. Außerdem erscheint eine Anleitung, wie man im folgenden Bild Kette und Schuß entwerfen kann.

Mit »Enter« kommt man ins nächste Bild. Oben links blinkt ein Pfeil und am unteren Bildrand erscheint die Anweisung: »WENN FERTIG '0'«. Drückt man nun eine Zifferntaste von 1 bis 6, so wird ein farbiger senkrechter Streifen gemalt, ein Kettfaden, und der Pfeil springt um eine Position nach rechts. Den Pfeil können Sie mit den Tasten »E« und »S« horizontal nach rechts und links schieben. Drücken Sie »X« oder »E«, so erscheint er am rechten Bildrand, und Sie können dann Schußfäden malen.

Auf dem Bildschirm haben bis zu 31 Kettfäden und bis zu 23 Schußfäden Platz, aber Sie müssen nicht alle einzeln malen. Beim Entwurf können Sie zwischen den einzelnen Fäden beliebig viele Lücken freilassen. Drücken Sie »0«, um zum Menü zu kommen. Mit »C« entfernen Sie die Lücken und der Computer malt Ihnen das komprimierte Muster voll. Schöner ist es aber, wenn man die Lücken nicht entfernt, sondern füllt (mit »D«). Der Computer füllt dann alle Lücken rechts von einem Kettfaden mit Fäden der gleichen Farbe, bis er auf einen vorhandenen andersfarbigen trifft. Rechts vom letzten Faden wird nicht mehr gefüllt. Analog geht es beim Schuß zu. Hier füllt der Computer von oben nach unten.

Wenn der Computer das komprimierte oder gefüllte Muster malt, können Sie den Vorgang mit »0« unterbrechen und ins Menü gelangen. Lassen Sie ihn aber zu Ende malen, so fragt er Sie, ob Sie die Farben ändern wollen. Bei »J« fragt er Sie, ob Sie Farbe 1 ändern wollen. So oft Sie nun »ENTER« drücken, ändert sich schlagartig die Farbe aller Fäden, die Sie mit Taste 1 gemalt haben, und zwar werden alle 15 Farben der Reihe nach und dann wieder von vorne beginnend angezeigt, bis Sie »N« drücken. Dann können Sie Farbe 2 und so weiter ändern.

Wenn Sie mit Ihrem Muster zufrieden sind, kommen Sie mit »ENTER« wieder ins Menü und von dort aus mit »F« zur Musterbeschreibung. Hier wird ausgedruckt, wieviele Fäden der ersten, zweiten, dritten Farbe und so weiter nebeneinander verwendet wurden.

Wenn Ihnen die Ausführung Ihres Entwurfs nicht gefallen hat, kommen Sie mit »E« wieder zu Ihrem unkomprimierten und ungefüllten Entwurf zurück und können ihn anders gestalten. Mit »A« kommen Sie vorher zur Farbtafel und können die Ausgangsfarben neu wählen, bevor Sie Ihren alten Entwurf ändern. Sie werden sicherlich bald feststellen, daß wenige Farben oft schöner sind als viele.

(Bernhard Decker)

10- 120	Initialisierung
130	Prescanning abschalten
140- 180	Hexcode für Fadencharakter und Farbe definieren
190- 300	Titelbild
310- 640	Farbtafel, Vorauswahl von 6 Farben
590- 640	Umdefinition Fadencharakter und Farbe
650- 730	Erklärungstafel
740- 850	Menü
860- 940	Routine Kettgröße feststellen. Lücken vor erstem Faden werden entfernt, dann feststellen, wo sich der letzte Faden auf der rechten Seite befindet. Z(I) enthält dann die Kette, B die Anzahl der Kettfäden.
950-1030	Routine Schußgröße feststellen (wie Kettgröße), Y(I) enthält den Schuß, B die Anzahl der Fäden
1040-1140	Kette komprimieren. Leerstellen werden entfernt und durch Fäden weiter rechts ersetzt. In 2630 wird der Kettenentwurf gespeichert
1150-1280	komprimiert den Schuß
1290-1350	Kette bildfüllend stellt fest, ob rechts vom letzten Kettfaden noch Platz ist, und wiederholt das Muster von links beginnend
1360-1420	Schuß bildfüllend macht dasselbe mit dem Schuß
1430-1790	Muster malen. Malt das komprimierte oder gefüllte Muster auf dem gesamten Bildschirm
1640-1790	gibt Gelegenheit, die ausgesuchten Farben im gemalten Muster zu ändern
1800-1930	gibt die Anzahl der benutzten Farben der nebeneinanderliegenden Fäden auf dem Bildschirm aus (für Kette und Schuß)
1940-1980	hält den Programmablauf an, bis »ENTER« gedrückt wird
1990-2060	malt auf der Erklärungstafel die ausgewählten Farben
2070-2310	Muster entwerfen. Routine, um senkrecht einzelne Linien in sechs ausgewählten Farben zu malen
2080	malt den in Z1(I) gespeicherten alten Entwurf, wenn w=1
2150-2310	einzelne Kettfäden malen
2320-2460	Muster füllen. Routine, um Lücken zwischen Einzelfäden mit Fäden der vorhergehenden Farbe zu füllen
2470-2610	Muster entwerfen. Routine, um waagerecht einzelne Linien in sechs ausgewählten Farben zu malen = Schußfäden
2620-2690	speichert Daten zum Kettentwurf, und zwar in »GK« die Anzahl verschiedener Farbstreifen, in »G1«(GK)« die Anzahl Fäden eines Farbstreifens, in »F1(GK)« die Farbe des Streifens
2700-2770	speichert Daten zum Schußentwurf, und zwar in »GS« die Anzahl verschiedener Farbstreifen, in »G2(GS)« die Anzahl Fäden eines Farbstreifens, in »F2(GK)« die Farbe des Streifens
2780-2880	malt einen Kettfaden
2890-2990	malt einen Schußfaden
3000	neuer Programmstart für neues Muster

Programmbeschreibung zu »Webmuster«

Z(32)	= Speicher Farbcode für Kettfäden
Z1(32)	= Zwischenspeicher für Z(32)
Y(24)	= Speicher Farbcode für Schußfäden
Y1(24)	= Zwischenspeicher für Y(24)
G1(32)	= Fadenzahl pro Farbstreifen in Kette
G2(24)	= Fadenzahl pro Farbstreifen in Schuß
F1(32)	= Farbcode der Farbstreifen in Kette
F2(24)	= Farbcode der Farbstreifen in Schuß
F(6)	= Speicher der gewählten Farben
S	= Schußzähler
K	= Kettzähler
C	= Variable für die zu malenden Zeichen
I, J, A, B	= Zähler
KEY, ST	= Variable in Keyroutine
FZ	= Anzahl der ausgewählten Farben
GK	= Anzahl Farbstreifen in Kette
GS	= Anzahl Farbstreifen in Schuß
W	= Flag, ist 0 bei Neuentwurf, 1 bei altem Entwurf
A1$, A2$, A3$, A4$	= Hexcode für Charakter
A5$	= Variable für Menüwahlvorschlag
A6$	= Rückgabevariable bei Farbänderung

Variablenliste zu »Webmuster«

```
10 ! * SCHOTTENMUSTER ENTWERFEN
80 DIM Z(32),Z1(32),Y(24),Y1
(24),G1(32),G2(24),F1(32),F2
(24),F(6)
90 GOTO 150 :: A1$ :: A2$ ::
 A3$ :: A4$ :: A5$ :: A6$
100 S :: K :: C :: I :: J ::
 KEY :: A :: B :: FZ :: GS :
: GK :: ST :: W
110 CALL KEY :: CALL SOUND :
: CALL CHAR :: CALL COLOR
120 CALL HCHAR :: CALL VCHAR
 :: CALL SCREEN :: CALL CLEAR
130 !@P-
140 REM CHARACTER-HEXCODE
150 A1$="FFFFFFFFFFFFFFFF"
160 A2$="7E7E7E7E7E7E7E7E"
170 A3$="00FFFFFFFFFFFF00"
180 CALL CHAR(97,A2$,98,A3$,
105,A2$,106,A3$):: CALL COLO
R(9,9,2,10,16,2)
190 ! TITEL
200 CALL CLEAR
210 DISPLAY AT(6,10):"SCHOTT
ENMUSTER"
220 DISPLAY AT(11,6):"DIESES
 PROGRAMM ZEIGT":TAB(6);"IHN
EN, WIE VON IHNEN"
230 DISPLAY AT(13,6):"ENTWOR
FENE MUSTER BEIM":TAB(6);"WE
BEN AUSSEHEN WERDEN."
240 DISPLAY AT(16,6):"DABEI
VERWENDET ES DIE": :TAB(10);
"LEINEN-BINDUNG,"
250 DISPLAY AT(20,6):"D.H.:
IM GEWEBE IST AB-"
260 DISPLAY AT(21,6):"WECHSE
LND SCHUSS UND":TAB(6);"KETT
E ZU SEHEN."
270 CALL HCHAR(1,1,98,32)::
CALL HCHAR(3,1,106,32)
280 CALL VCHAR(1,2,97,24)::
CALL VCHAR(1,4,105,24)
290 CALL HCHAR(1,4,98):: CAL
L HCHAR(3,2,106)
300 GOSUB 1950
310 REM FARBTAFEL
320 CALL CLEAR :: I=1
330 FOR J=2 TO 12 STEP 2
340 CALL COLOR(8+I,J,J+1)::
I=I+1
350 CALL CHAR(88+4*J,A1$,92+
4*J,A4$)
360 NEXT J
370 DISPLAY AT(1,1):"FARBTAF
EL:"
380 B=1 :: A=2
390 FOR I=1 TO 12
400 A=A+1 :: IF A>10 THEN A=
3 :: B=15
410 DISPLAY AT(A,B):USING "#
#":I
420 CALL HCHAR(A,B+5,92+4*I,
7)
430 NEXT I
440 DISPLAY AT(8,15):"13 VIO
LETT"
450 DISPLAY AT(9,15):"14 GRA
U" :: DISPLAY AT(10,15):"15
WEISS"
460 DISPLAY AT(13,1):"WAEHLE
N SIE 6 FARBEN!":"NACH MUSTE
RENTWURF AENDERBAR"
470 FZ=6
480 CALL HCHAR(15,1,32,96)
490 DISPLAY AT(16,1):"IHRE F
ARBWAHL:"
500 FOR I=1 TO 6
510 DISPLAY AT(17+I,1):"FARB
E ";STR$(I);":";I
520 NEXT I
530 FOR I=1 TO 6
540 ACCEPT AT(17+I,10)VALIDA
TE(DIGIT)SIZE(-2):J
550 IF J=0 THEN 580
560 IF J>15 THEN 540 ELSE F(
I)=J+1
570 NEXT I
580 FZ=I-1 :: CALL CLEAR
590 FOR I=1 TO FZ !FARBCHARA
CTER INITIIEREN
600 CALL COLOR(8+I,F(I),2)
610 CALL CHAR(88+8*I,A1$)
620 CALL CHAR(89+8*I,A2$)
630 CALL CHAR(90+8*I,A3$)
640 NEXT I
650 REM ERKLAERUNGSEITE
660 GOSUB 2000 :: A5$="E"
670 DISPLAY AT(10,1):"AUF DE
M FOLGENDEN BILD SEHEN"
```

Listing zu »Webmuster«

```
680 DISPLAY AT(11,1):"SIE EI
NEN PFEIL, DEN SIE MIT"
690 DISPLAY AT(12,1):"DEN PF
EILTASTEN >E,S,D,X<":"BEWEGE
N KOENNEN."
700 DISPLAY AT(15,1):"DRUECK
EN SIE EINE ZIFFERN-":"TASTE
, SO MALT DER COMPUTER":"EIN
EN FADEN IN DIESER FARBE."
710 DISPLAY AT(19,1):"LUECKE
N IM ENTWURF KANN DER COMPUT
ER FUELLEN.":"WOLLEN SIE DAS
 ENTWERFEN":"BEENDEN, TASTE
>0< DRUECKEN!"
720 GOSUB 1950
730 GOTO 2080
740 REM MENUE
750 CALL CLEAR
760 DISPLAY AT(1,1):"* KOMPR
IMIEREN = DIE FAEDEN":"  WER
DEN ZUSAMMENGESCHOBEN,":"  U
. BILDFUELLEND WIEDERHOLT"
770 DISPLAY AT(7,1):"* FUELL
EN = LEERAEUME WER-":"  DEN
GLEICHFARBIG GEFUELLT"
780 DISPLAY AT(12,1):"WOLLEN
 SIE:"
790 DISPLAY AT(14,1):"A: STA
RTFARBEN AENDERN?":"B: NEUES
 MUSTER ENTWERFEN?":"C: MUST
ER KOMPRIMIEREN?":"D:
 FUELLEN?"
800 DISPLAY AT(18,1):"E: ZUM
 ENTWURF ZURUECK?":"F: MUSTE
RBESCHREIBUNG?":"G: ZUR ERKL
AERUNGSTAFEL?"
810 DISPLAY AT(21,1):"H: BEE
NDEN?"
820 DISPLAY AT(24,1):"IHRE W
AHL: ";A5$
830 ACCEPT AT(24,12)SIZE(-1)
VALIDATE("ABCDEFGH")BEEP:A5$
840 KEY=ASC(A5$)-64
850 ON KEY GOTO 320,3010,105
0,2330,2080,1810,660,3020
860 REM KETTENGROESSE FESTST
ELLEN
870 FOR I=1 TO 30 :: IF Z(I)
>0 THEN 890
880 NEXT I
890 IF I=1 THEN 910
900 FOR J=1 TO 30-I :: Z(J)=
Z(J+I-1):: Z(J+I-1)=0 :: NEX
T J !SCHIEBEREGISTER, WENN V
ORDERE KETTE LEER
910 FOR I=30 TO 1 STEP -1
920 IF Z(I)>0 THEN B=I :: GO
TO 940
930 NEXT I
940 RETURN
950 REM SCHUSSGROESSE FESTST
ELLEN
960 FOR I=1 TO 23 :: IF Y(I)
>0 THEN 980
970 NEXT I
980 IF I=1 THEN 1000
990 FOR J=1 TO 23-I :: Y(J)=
Y(J+I-1):: Y(J+I-1)=0 :: NEX
T J
1000 FOR I=23 TO 1 STEP -1
1010 IF Y(I)>0 THEN B=I :: G
OTO 1030
1020 NEXT I
1030 RETURN
1040 REM KETTE KOMPRIMIEREN
1050 GOSUB 870
1060 A=0
1070 FOR I=1 TO B
1080 A=A+1 :: IF A>B THEN 11
20
1090 IF Z(A)=0 THEN 1080
1100 Z(I)=Z(A):: IF A>I THEN
 Z(A)=0
1110 NEXT I
1120 GOSUB 2630
1130 IF KEY=6 THEN 1160
1140 GOSUB 1300
1150 REM SCHUSS KOMPRIMIEREN
1160 GOSUB 950
1170 A=0
1180 FOR I=1 TO B
1190 A=A+1 :: IF A>B THEN 12
30
1200 IF Y(A)=0 THEN 1190
1210 Y(I)=Y(A):: IF A>I THEN
 Y(A)=0
1220 NEXT I
1230 GOSUB 2710
1240 IF KEY=6 THEN 1820
1250 GOSUB 1370
1260 I,J=0 :: GOSUB 1440
1270 GOSUB 1950
1280 GOTO 750
1290 REM KETTE BILDFUELLEND
1300 A=0
1310 FOR I=1 TO 32
1320 IF Z(I)>0 THEN 1340
1330 A=A+1 :: Z(I)=Z(A)
1340 NEXT I
1350 RETURN
1360 REM SCHUSS BILDFUELLEND
1370 A=0
1380 FOR I=1 TO 24
1390 IF Y(I)>0 THEN 1410
1400 A=A+1 :: Y(I)=Y(A)
1410 NEXT I
1420 RETURN
1430 REM MUSTER MALEN
1440 A5$="F" :: CALL CLEAR :
```

Listing zu »Webmuster« (Fortsetzung)

```
: PRINT "UNTERBRECHEN MIT TA
STE >Ø<"
1450 FOR S=1 TO 23 !S=SCHUSS
1460 FOR K=1 TO 32 !K=KETTE
1470 CALL KEY(Ø,KEY,ST)
1480 IF KEY=48 THEN 1790
1490 C=89+8*Z(K)!KETTE SICHT
BAR
1500 CALL HCHAR(S,K,C)
1510 K=K+1
1520 C=90+8*Y(S)!SCHUSS SICH
TBAR
1530 CALL HCHAR(S,K,C)
1540 NEXT K
1550 S=S+1
1560 FOR K=1 TO 32
1570 C=90+8*Y(S)!SCHUSS SICH
TBAR
1580 CALL HCHAR(S,K,C)
1590 K=K+1
1600 C=89+8*Z(K)!KETTE SICHT
BAR
1610 CALL HCHAR(S,K,C)
1620 NEXT K
1630 NEXT S
1640 REM FARBE NACHTRAEGLICH
 AENDERN
1650 DISPLAY AT(24,13):"FARB
E AENDERN? J"
1660 ACCEPT AT(24,28)VALIDAT
E("JN")SIZE(-1):A6$
1670 IF A6$="N" THEN 1780
1680 FOR I=1 TO FZ
1690 DISPLAY AT(24,11):"FARB
E ";STR$(I);" AENDERN? J"
1700 ACCEPT AT(24,28)VALIDAT
E("JN")SIZE(-1):A6$
1710 IF A6$="N" THEN 1770
1720 FOR J=1 TO 16
1730 F(I)=J :: CALL COLOR(8+
I,J,2)
1740 ACCEPT AT(24,28)VALIDAT
E("JN")SIZE(-1):A6$
1750 IF A6$="N" THEN 1770
1760 NEXT J :: GOTO 1720
1770 NEXT I
1780 CALL HCHAR(24,13,32,19)
1790 RETURN
1800 REM FADENANGABEN AUSDRU
CKEN
1810 A5$="B" :: IF GK=Ø THEN
 1050
1820 CALL CLEAR
1830 DISPLAY AT(1,1):"IHRE K
ETTE BESTEHT AUS:"
1840 FOR I=1 TO MIN(GK,22)
1850 DISPLAY AT(2+I,1):G1(I)
;"FAEDEN"
1860 CALL HCHAR(2+I,14,90+8*
F1(I),1Ø)
1870 NEXT I
1880 DISPLAY AT(12,1):"IHR S
CHUSS BESTEHT AUS:"
1890 FOR I=1 TO MIN(GS,11)
1900 DISPLAY AT(13+I,1):G2(I
);"FAEDEN"
1910 CALL HCHAR(13+I,14,90+8
*F2(I),1Ø)
1920 NEXT I
1930 GOTO 1270 !ZUM MENUE
1940 REM SCHALTEN
1950 DISPLAY AT(24,22)BEEP:"
>ENTER<"
1960 CALL KEY(Ø,KEY,ST)
1970 IF KEY<>13 THEN 1960
1980 RETURN
1990 REM GEWAEHLTE FARBEN MA
LEN
2000 DISPLAY AT(1,1)ERASE AL
L:"IHRE FARBEN:"
2010 FOR I=1 TO FZ
2020 DISPLAY AT(2+I,1):"FARB
E";I :: DISPLAY AT(2+I,8):":
"
2030 CALL HCHAR(2+I,12,90+8*
I,11)
2040 DISPLAY AT(2+I,22):"TAS
TE";I
2050 NEXT I
2060 RETURN
2070 REM MUSTER ENTWERFEN
2080 IF W=Ø THEN 2150 ELSE D
ISPLAY AT(1,1)ERASE ALL:"BIS
HERIGER ENTWURF:"
2090 FOR I=1 TO 24 :: Z(I)=Z
1(I):: Y(I)=Y1(I):: NEXT I
2100 FOR I=25 TO 32 :: Z(I)=
Z1(I):: NEXT I
2110 FOR K=1 TO 30 :: IF Z(K
)>Ø THEN GOSUB 2790
2120 NEXT K
2130 FOR S=1 TO 23 :: IF Y(S
)>Ø THEN GOSUB 2900
2140 NEXT S :: GOTO 2170
2150 CALL CLEAR :: W=1
2160 REM KETTE
2170 A5$="D" :: DISPLAY AT(2
4,13):"WENN FERTIG >Ø<!"
2180 CALL CHAR(92,"101010FE7
C3810")!PFEIL NACH UNTEN
2190 J=2 :: CALL VCHAR(1,32,
32,24)
2200 CALL SOUND(50,200,9)
2210 CALL KEY(Ø,KEY,ST)
2220 CALL HCHAR(1,2,32,31)::
 CALL HCHAR(1,J,92)
2230 IF ST=Ø THEN 2210
2240 IF KEY=69 OR KEY=88 THE
N 2480
```

Listing zu »Webmuster« (Fortsetzung)

```
2250 IF KEY=83 THEN J=MAX(2,
J-1):: GOTO 2200
2260 IF KEY=68 THEN J=MIN(31
,J+1):: GOTO 2200
2270 IF KEY<48 OR KEY>48+FZ
THEN 2210
2280 IF KEY=48 THEN 750
2290 KEY=KEY-48 :: K=J-1 ::
Z(K),Z1(K)=KEY
2300 GOSUB 2790 !KETTFADEN M
ALEN
2310 J=MIN(31,J+1):: GOTO 22
00
2320 REM MUSTER FUELLEN
2330 GOSUB 870 !KETTGROESSE
2340 FOR I=1 TO B-1
2350 IF Z(I)>0 THEN 2370
2360 Z(I)=Z(I-1)
2370 NEXT I
2380 GOSUB 2630
2390 GOSUB 1300
2400 GOSUB 950 !SCHUSSGROESS
E
2410 FOR I=1 TO B-1
2420 IF Y(I)>0 THEN 2440
2430 Y(I)=Y(I-1)
2440 NEXT I
2450 GOSUB 2710
2460 GOTO 1250
2470 REM SCHUSS ENTWERFEN
2480 CALL CHAR(92,"040C1C3F1
C0C04")!PFEIL NACH LINKS
2490 I=2 :: CALL HCHAR(1,1,3
2,32)
2500 CALL SOUND(50,175,9)
2510 CALL KEY(0,KEY,ST)
2520 CALL VCHAR(1,32,32,24):
: CALL VCHAR(I,32,92)
2530 IF ST=0 THEN 2510
2540 IF KEY=83 OR KEY=68 THE
N 2180
2550 IF KEY=69 THEN I=MAX(2,
I-1):: GOTO 2500
2560 IF KEY=88 THEN I=MIN(24
,I+1):: GOTO 2500
2570 IF KEY<48 OR KEY>48+FZ
THEN 2510
2580 IF KEY=48 THEN 750
2590 KEY=KEY-48 :: S=I-1 ::
Y(S),Y1(S)=KEY
2600 GOSUB 2900
2610 I=MIN(24,I+1):: GOTO 25
00
2620 REM ANGABEN ZUR KETTE S
PEICHERN
2630 GK=0
2640 FOR I=1 TO 32
2650 IF Z(I)=0 THEN 2690
2660 IF Z(I)<>Z(I-1)THEN GK=
GK+1 :: A=1 ELSE A=A+1
2670 G1(GK)=A :: F1(GK)=Z(I)
!G1=FADENZAHL,F1=FARBE,GK=AN
ZAHL FARBSTREIFEN
2680 NEXT I
2690 RETURN
2700 REM ANGABEN ZUM SCHUSS
SPEICHERN
2710 GS=0
2720 FOR I=1 TO 24
2730 IF Y(I)=0 THEN 2770
2740 IF Y(I)<>Y(I-1)THEN GS=
GS+1 :: A=1 ELSE A=A+1
2750 G2(GS)=A :: F2(GS)=Y(I)
2760 NEXT I
2770 RETURN
2780 REM EINZELNEN KETTFADEN
 MALEN
2790 FOR S=1 TO 23
2800 IF Y(S)>0 THEN 2830
2810 C=89+8*Z(K)!KETTE SICHT
BAR
2820 CALL HCHAR(S+1,K+1,C)::
 GOTO 2870
2830 IF K/2<>INT(K/2)AND S/2
<>INT(S/2)THEN 2810
2840 IF K/2=INT(K/2)AND S/2=
INT(S/2)THEN 2810
2850 C=90+8*Y(S)!SCHUSS SICH
TBAR
2860 CALL HCHAR(S+1,K+1,C)
2870 NEXT S
2880 RETURN
2890 REM EINZELNEN SCHUSSFAD
EN MALEN
2900 FOR K=1 TO 30
2910 IF Z(K)>0 THEN 2940
2920 C=90+8*Y(S)!SCHUSS SICH
TBAR
2930 CALL HCHAR(S+1,K+1,C)::
 GOTO 2980
2940 IF K/2<>INT(K/2)AND S/2
=INT(S/2)THEN 2920
2950 IF K/2=INT(K/2)AND S/2<
>INT(S/2)THEN 2920
2960 C=89+8*Z(K)!KETTE SICHT
BAR
2970 CALL HCHAR(S+1,K+1,C)
2980 NEXT K
2990 RETURN
3000 REM NEUES MUSTER
3010 RUN 10
3020 END
```

Listing zu »Webmuster« (Schluß)

»qualimetric®« der letzte Schliff

Qualität bringt Sicherheit, und Sicherheit macht sich bezahlt. Besonders bei Speichermedien, die der Leistung Ihres wertvollen Computers entsprechen sollen. Dazu gehören Produkte, an denen der Hersteller nicht spart. BASF Speichermedien bekommen den letzten Schliff, und wir wissen mehr als andere, daß jedes Medium seine speziellen Schliff braucht: Qualität nach Maß. Achten Sie auf das »qualimetric«-Zeichen, wenn Sie Ihre Computer mit zuverlässigen Speichermedien versorgen wollen. Es lohnt sich.

® qualimetric

This collection is for all TI99/4A user.
If you like TI99 computer join at International Forum present on AtariAge.com site.
Greetings from Italy. Astharot

BASF